Jet Propulsion Laboratory
California Institute of Technology

Kigen: Supporting Spacecraft Design through Applied Artificial Intelligence

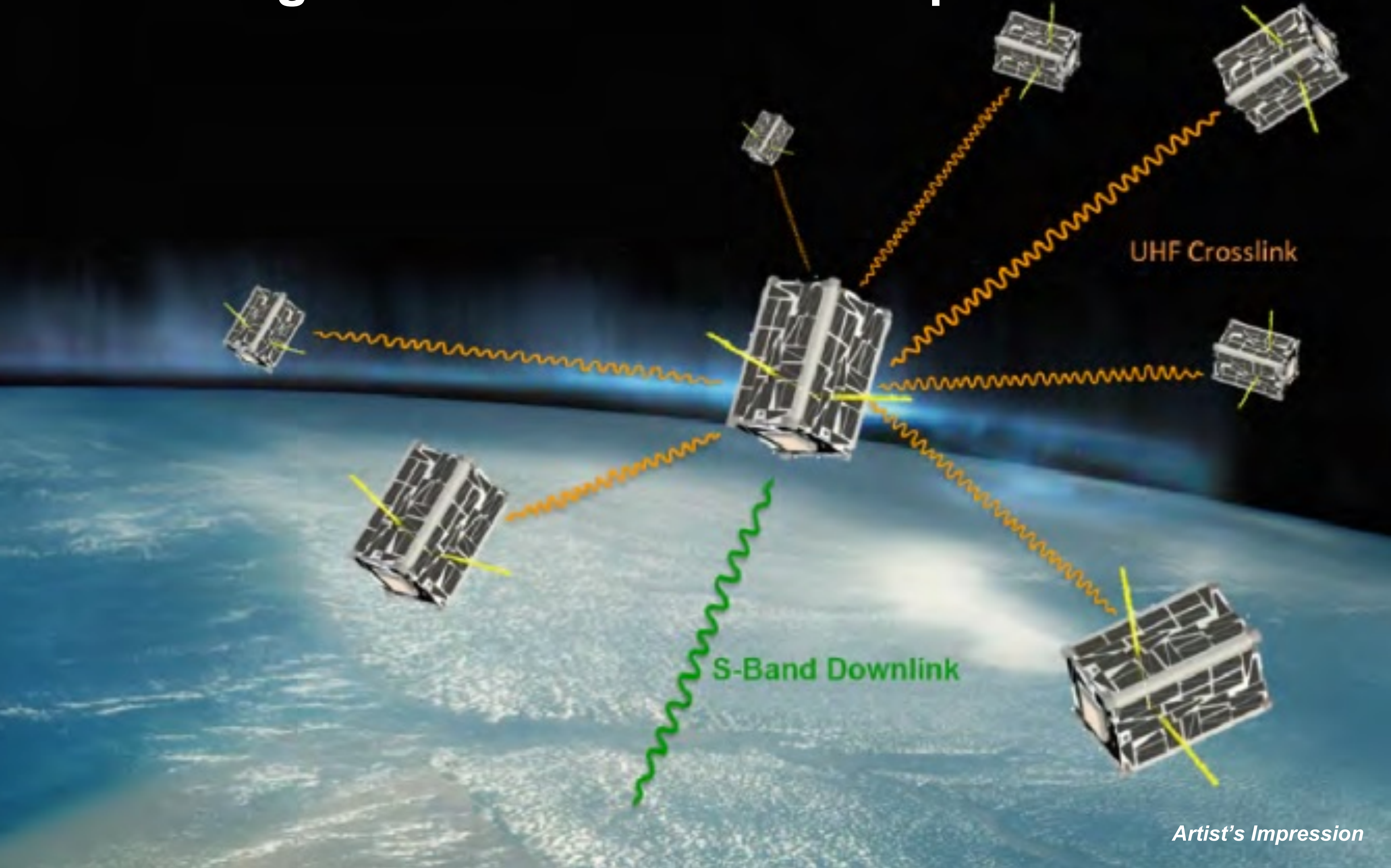
Dr. Sebastian J. I. Herzig

Jet Propulsion Laboratory, California Institute of Technology

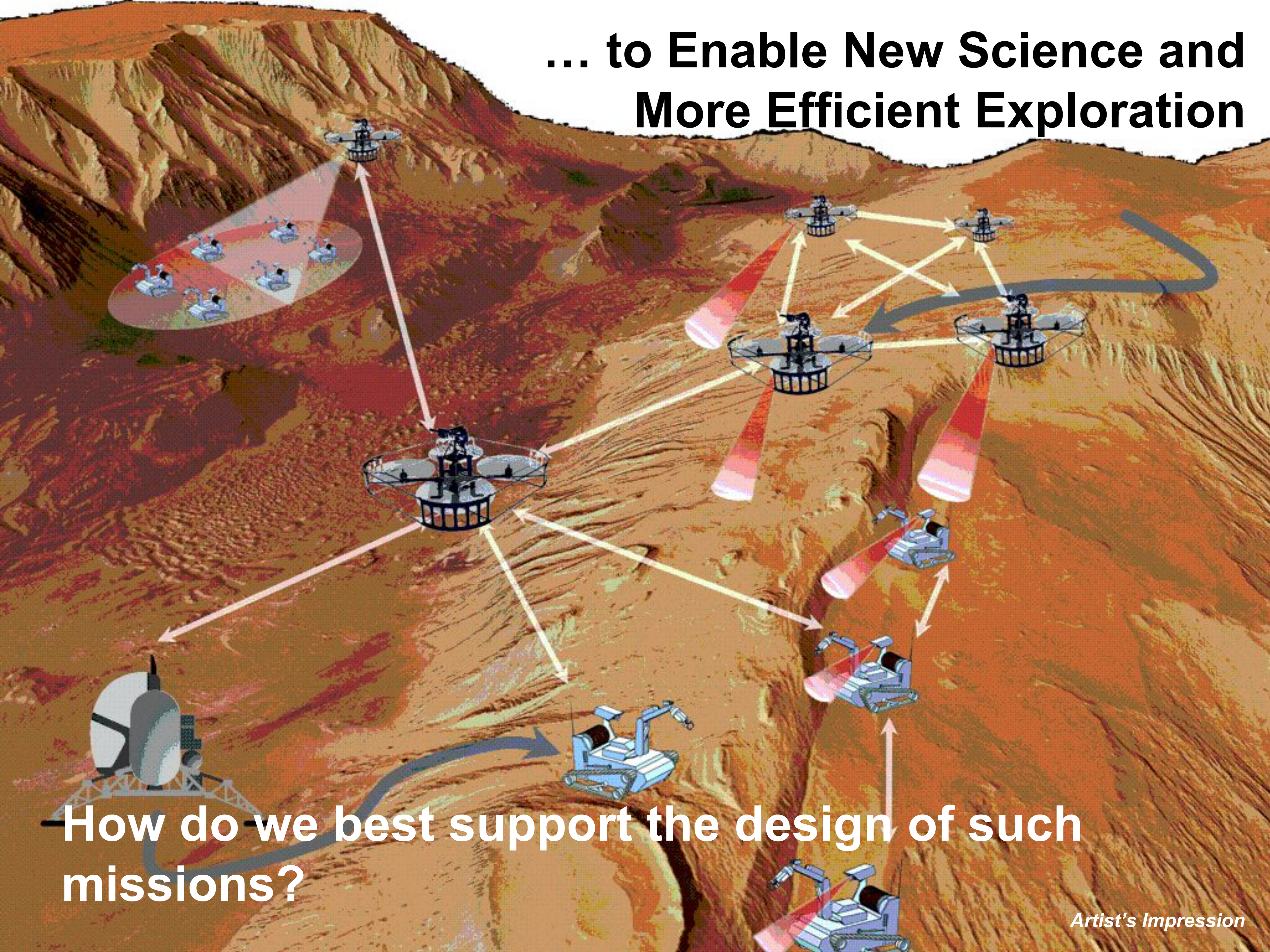
1 July 2018

Designing with AI, Design Computing and Cognition Conference (DCC'18), Milan, Italy

NASA Scientists are Envisioning Missions Involving Fleets of 10s of Small Spacecraft…



… to Enable New Science and More Efficient Exploration



How do we best support the design of such missions?

Artist's Impression

Motivating Example

Spacecraft-Based Radio Interferometry

Source: http://www.passmyexams.co.uk/GCSE/physics/images/radio-telescopes-outdoors.jpg

Radio interferometers:

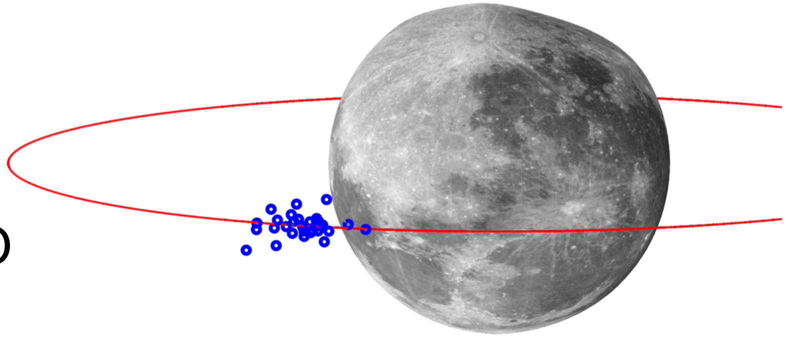
- Radio telescopes consisting of multiple low-frequency antennas
- Achieve the same angular resolution as that of a single telescope with the same aperture

➔ Typically ground-based

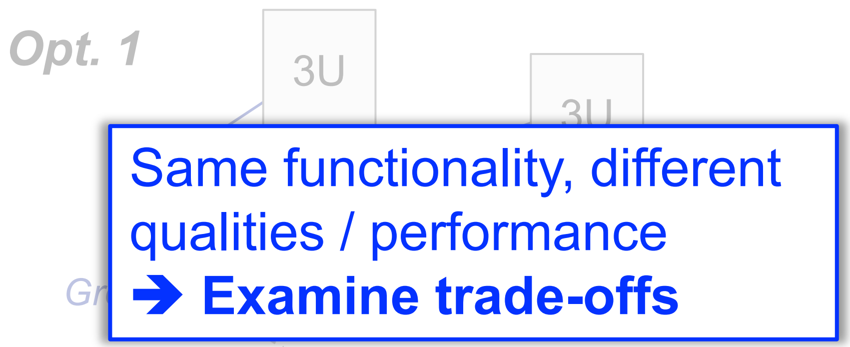
Want to do this in space:

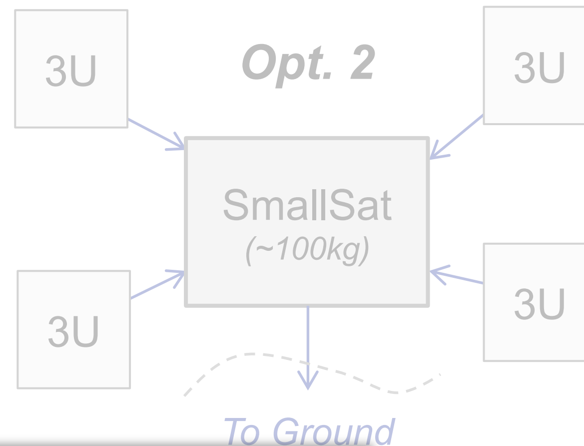
- Frequencies < 30Mhz blocked by ionosphere
- Cluster of spacecraft (3 – 50) functioning as interferometer in LLO

➔ CubeSats or SmallSats are promising enablers for this



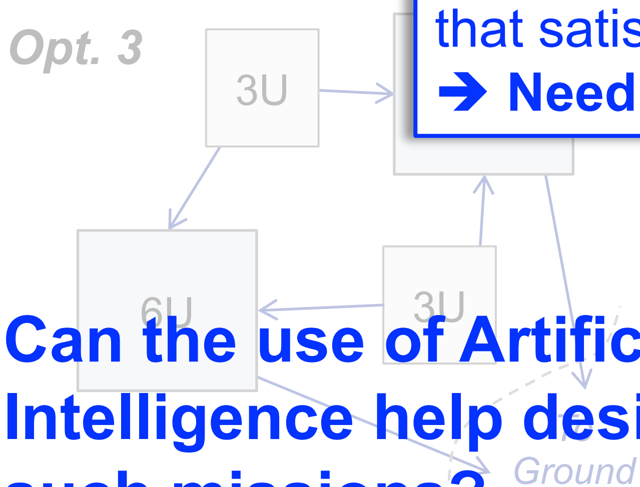
Which Architecture is Optimal?





Same functionality, different qualities / performance
➔ **Examine trade-offs**

Very large number of architectures that satisfy mission objectives
➔ **Need automation**



limited
...ge: transmit very
large amounts of data (TBs) to Earth

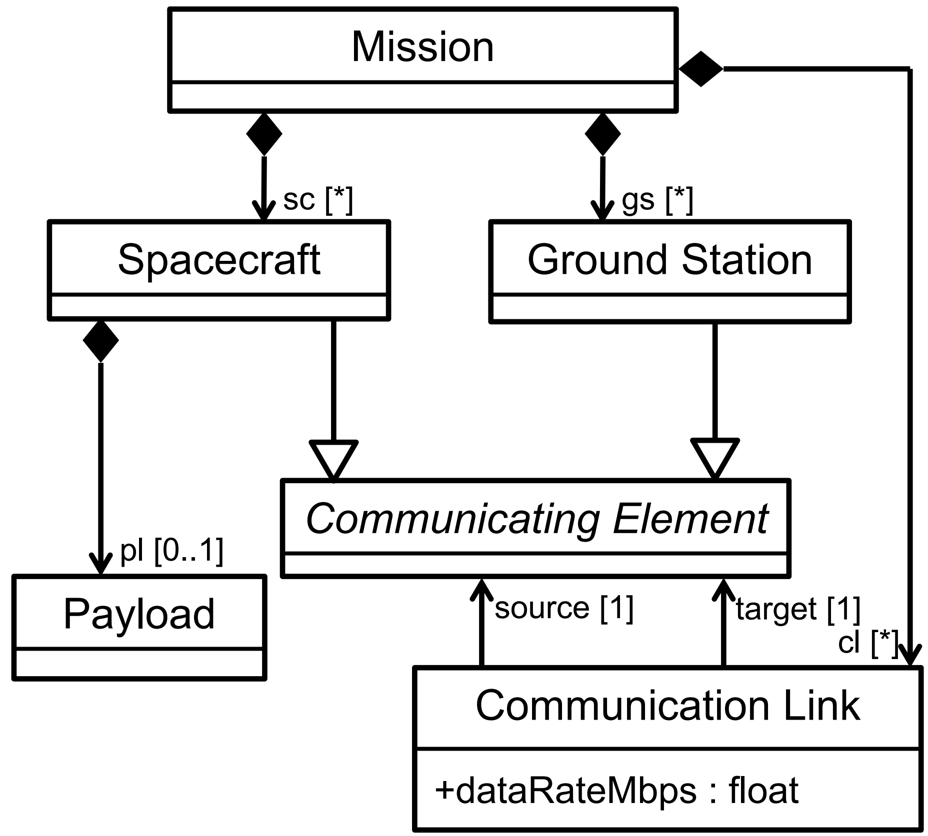
- How many spacecraft?
- Are all ...
  payloa...
- Who communicates with Earth?
- What frequency bands? Multi-hop?
- Optimal w.r.t. cost? Science value?

Functional allocation is key
➔ **Synthesis problem**

Can the use of Artificial Intelligence help design such missions?

Abstracting the Problem Domain

- Domain model
  - Concepts and behavior
  - Attributes and parametric relations
  - Associations & other relations
  - ➔ Describes a **universe of discourse**: many models in domain
  - ➔ Describes structural part of the problem



- Typically with addl. well-formedness constraints, e.g.:

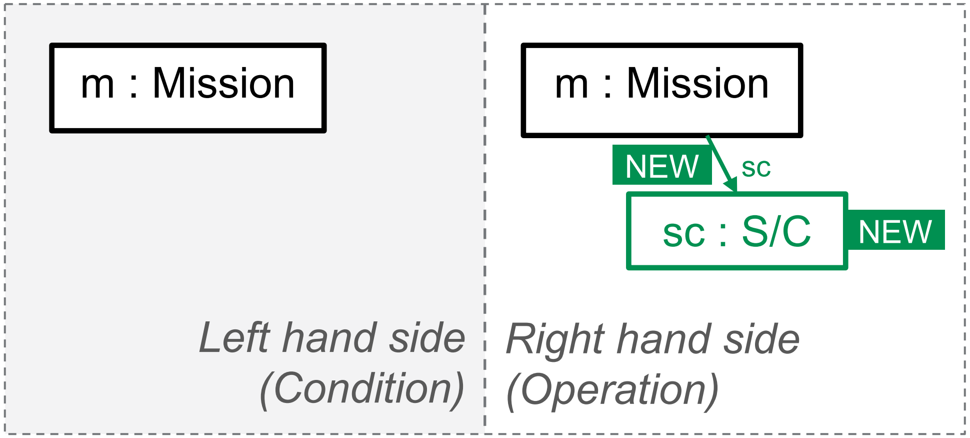
"All spacecraft must (transitively) be connected to at least one ground station through a communication link"

Any well-formed model in the domain is a candidate solution

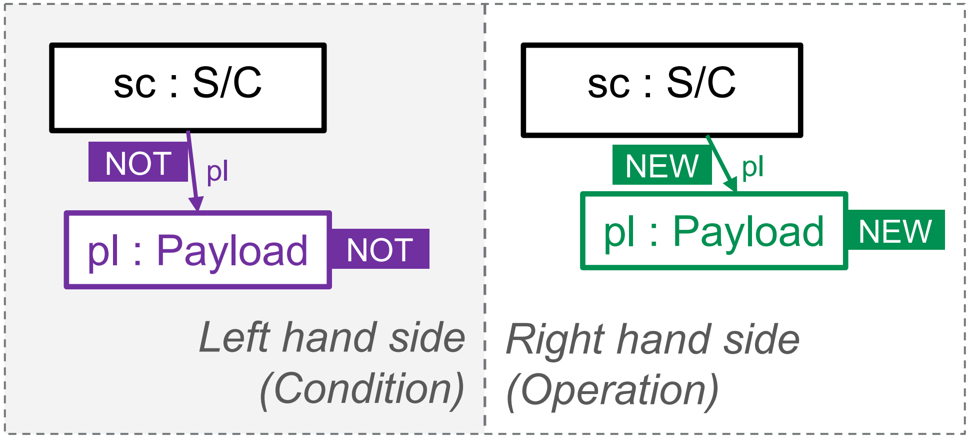
Rule-Based Model Generation

Model Transformation Rules as Enablers for Evolving Solutions

- Model transformation rules as operations for creating and / or evolving a given design / model in domain
  - **LHS**: Condition for match in input model (e.g., *"find an element of type Mission"*)
  - **RHS**: Operation to be performed (object creation, modification, deletion)
- Here: *endogenous, in-place* transformations
- Can also encode design heuristics this way!



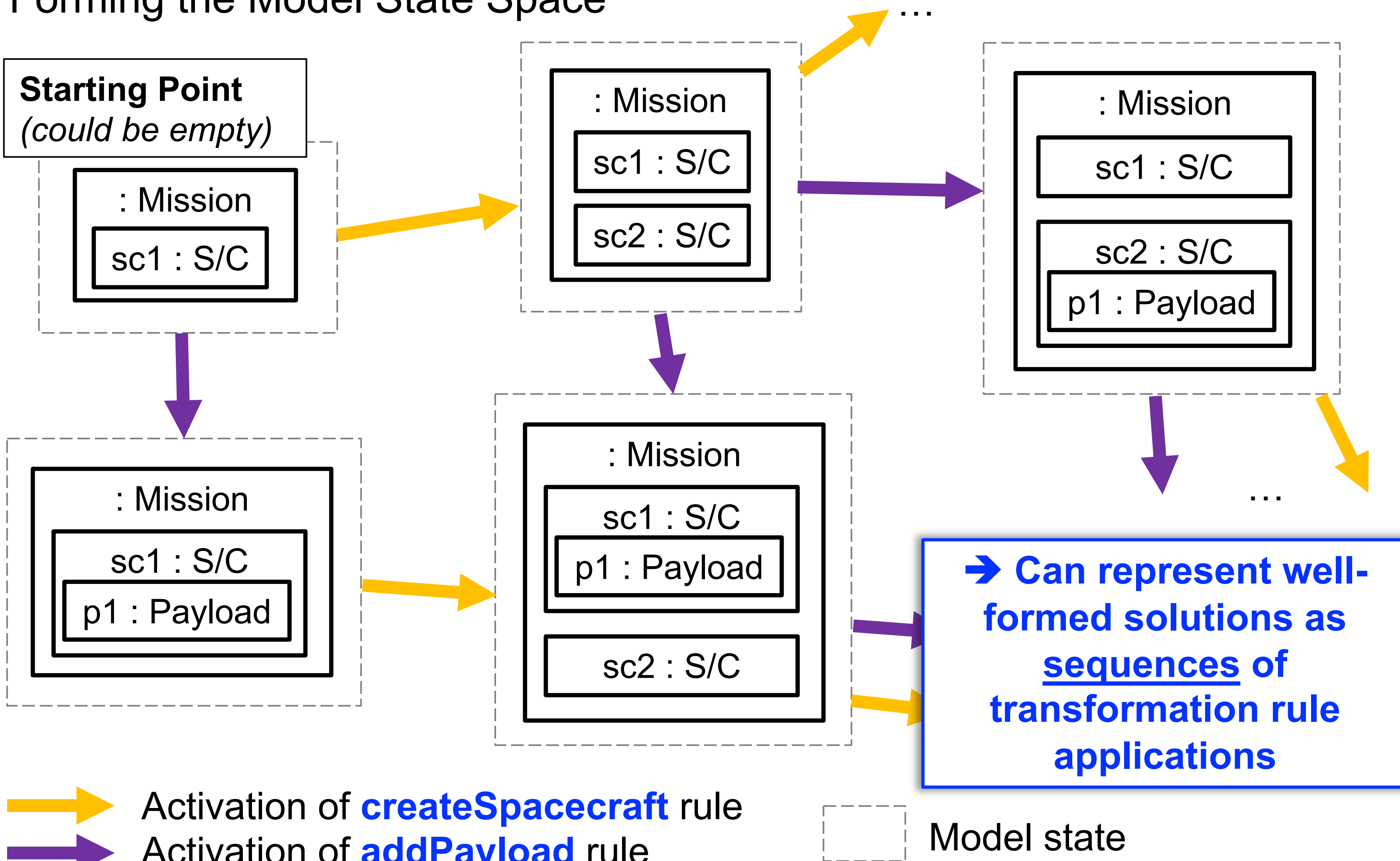
Rule "createSpacecraft"



Rule "addPayload"

Model-Transformation-Based Exploration

Forming the Model State Space



Evolving a Population of Models

Example: Using Genetic Algorithms to find Pareto-Optimal Solutions

Determined through analysis / simulation

Crossover

Individual x: *(Selection from population)*	**Add Spacecraft**	**Add X-Band Comm**	**Add Spacecraft**	**Add Comm Link**	***fitness=0.6***
					(Obj. Fct. Values)
Individual y:	**Add Spacecraft**	**Add Ka-Band Comm**	**Add Payload**	**Add Spacecraft**	***fitness=0.5***

Here, individuals are **sequences of transformation rule activations**
→ Each genome in population is a variable with set of trafo rules as range

New: *(Recombined individual in next generation)*

Add Ka-Band Comm ***fitness=0.9***

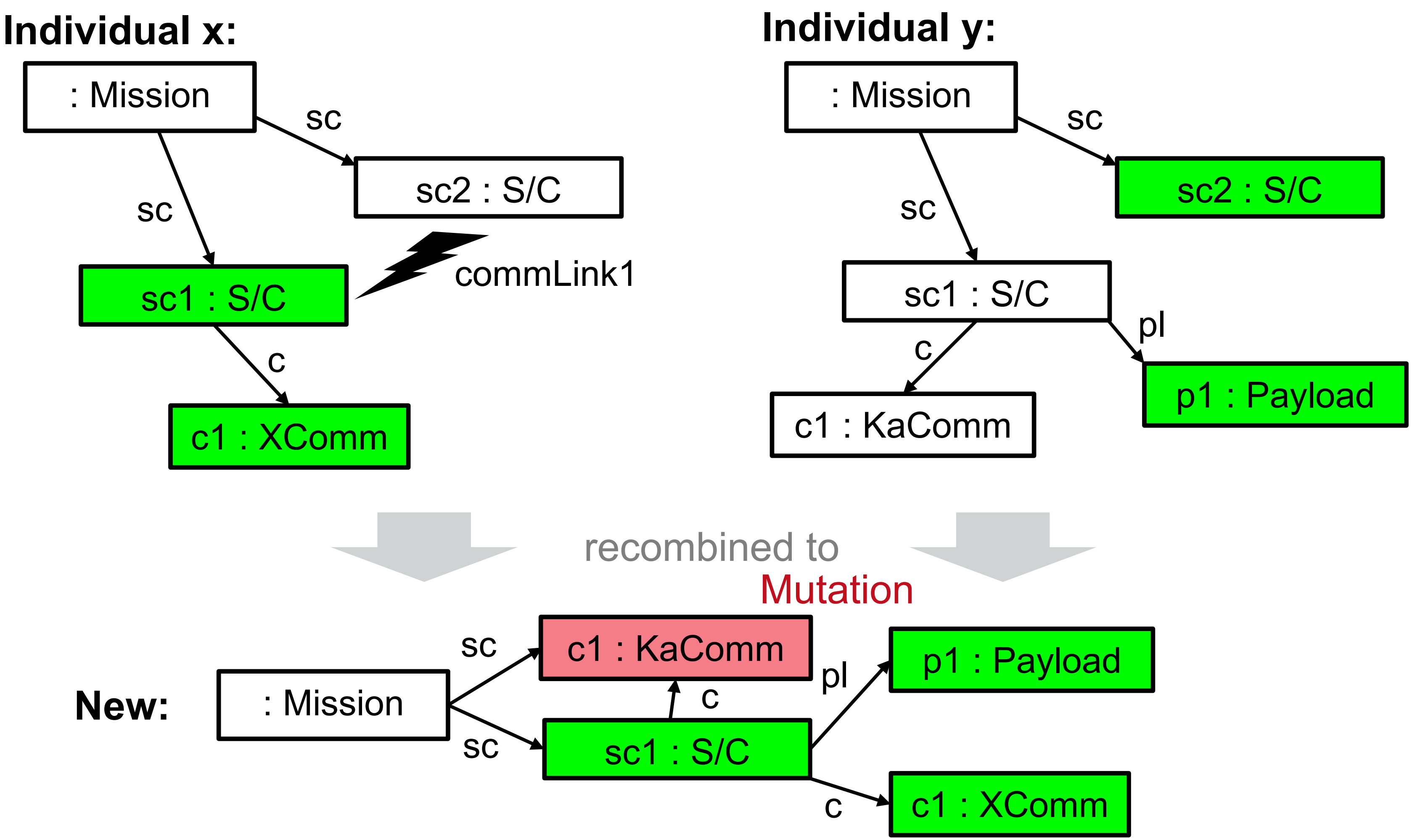
Mutation

Could also be a "placeholder" transformation (= rule "do nothing")

Can also use other Optimization Techniques, e.g.: Hill Climbing

Evolving a Population of Models

Illustrative Example



The Kigen Modeling Language (KigenML)

Design Problem Definition in KigenML

```
/* Domain description */
type ConstellationMission {
     var spacecraft : Spacecraft [ 2 .. 50 ]

     fun cost = { …// Some complex function
     fun benefit = { … // Another function

     constraint atLeast2SC: filter(
          spacecraft,
          (s : Spacecraft)
              => s.payload != null
        ).size > 1)
}

type Spacecraft {
     var payload : Payload [ 0 .. 1 ]
}

abstract type Payload

type InterferometryPayload
     extends Payload
```

```
/* Initial conditions */
model myMission of ConstellationMission {
     model sc1 of Spacecraft in spacecraft
     model sc2 of Spacecraft in spacecraft {
          model p1 of InterferometryPayload
              in payload
     }
}

/* Define optimization criteria */
minimize myMission.cost
maximize myMission.benefit

solve optimize using 'nsgaii' 30 times
     with
          populationSize 100,
          maxGenerations 20,
          maxSolutionLength 15
```

Able to derive domain model, rules & compile executable optimization problem from a Kigen program

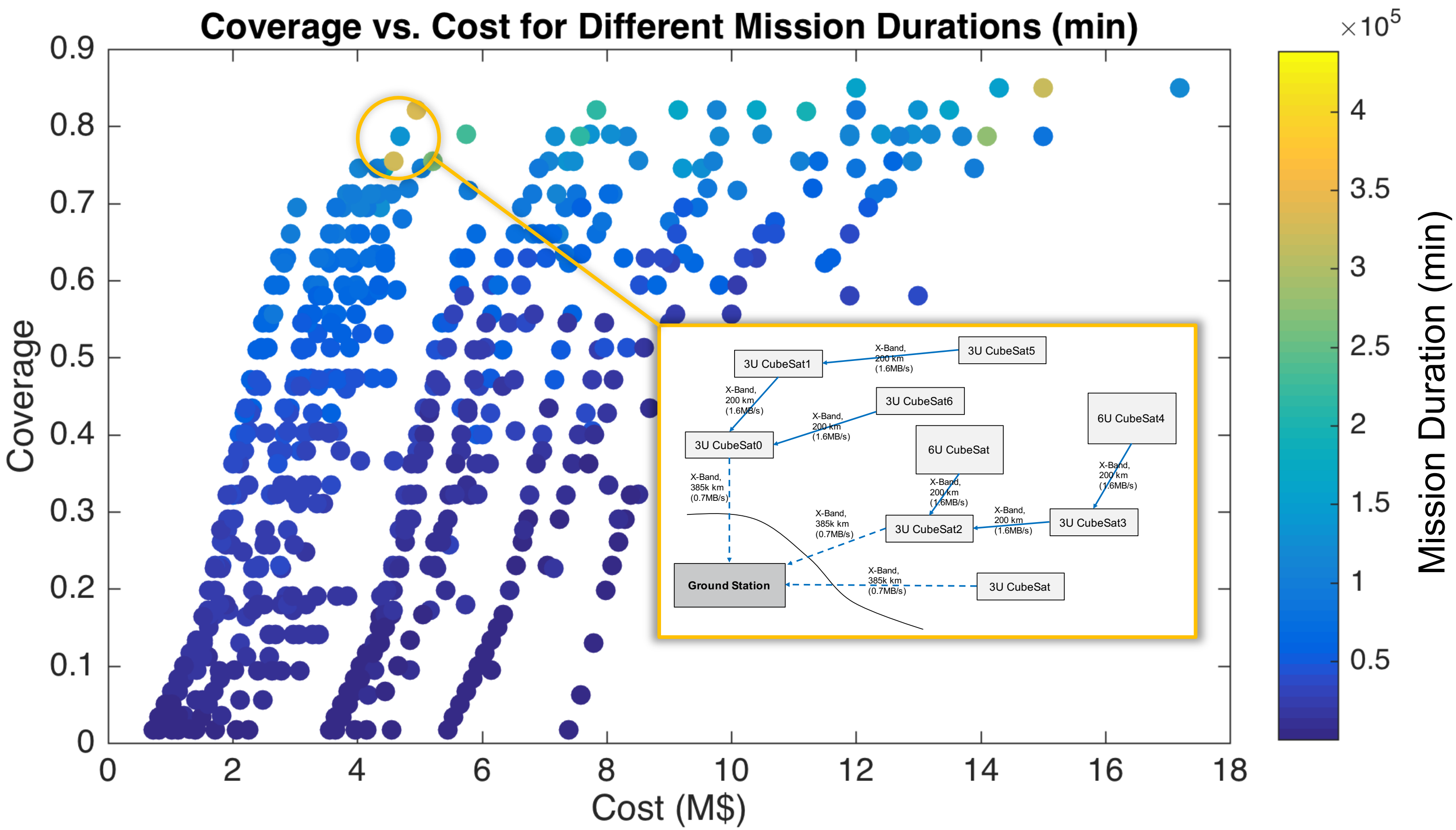
Implementation

Open Source Technologies Used in Implementation

- Kigen Modeling Language
  → Xtext
- Representation of Domain
  → Ecore / Eclipse EMF + OCL
- Exploration Rules
  → Henshin
- Analysis Models / Fitness Functions
  → Java
- Optimization Using Genetic Algorithms
  → MOMoT, MOEA

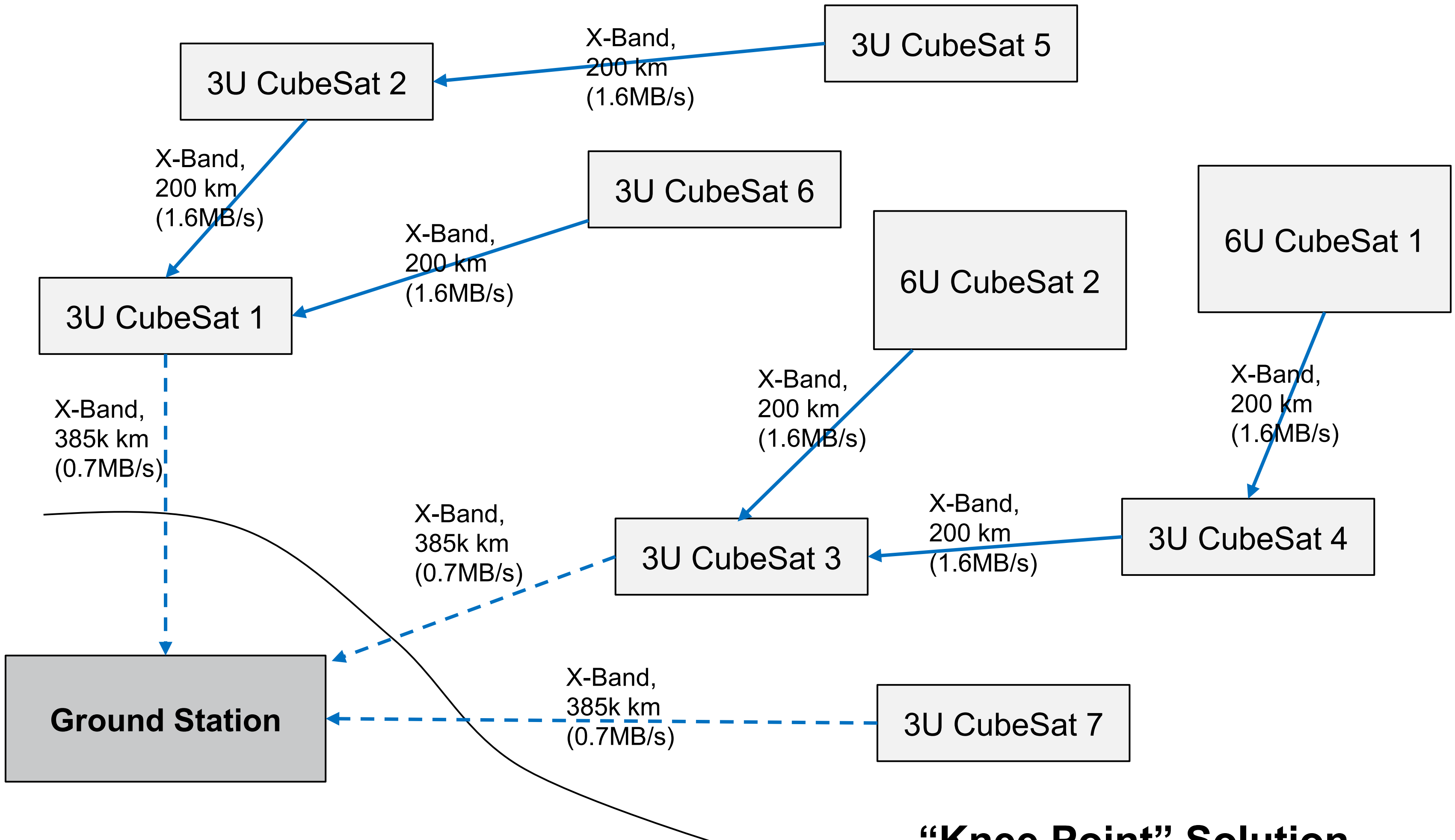
Results from Application to Case Study

Visualization of Trade Space



Results from Application to Case Study

Knee Point Solution



"Knee Point" Solution
$4.7M, ~0.79 coverage (10h observation)

Conclusions

- Demonstrated the use of AI methods for supporting design & systems engineers in exploring a highly complex trade space
  - Able to generate diverse set of alternatives in a reasonable timeframe
  - Solutions determined based on abstract description of problem
  - Enables designers to focus on **analysis**, and consider more options
- The generated candidate solutions can help spark creativity, but the method doesn't replace a team of engineers
  - Meant to ***support*** designers in the creative process
  - Analyzing results of optimizer can reveal missing information if results don't seem sensible
- Good performance for problems with limited scope, but should investigate methods for automatically dividing into sub-problems

Jet Propulsion Laboratory
California Institute of Technology

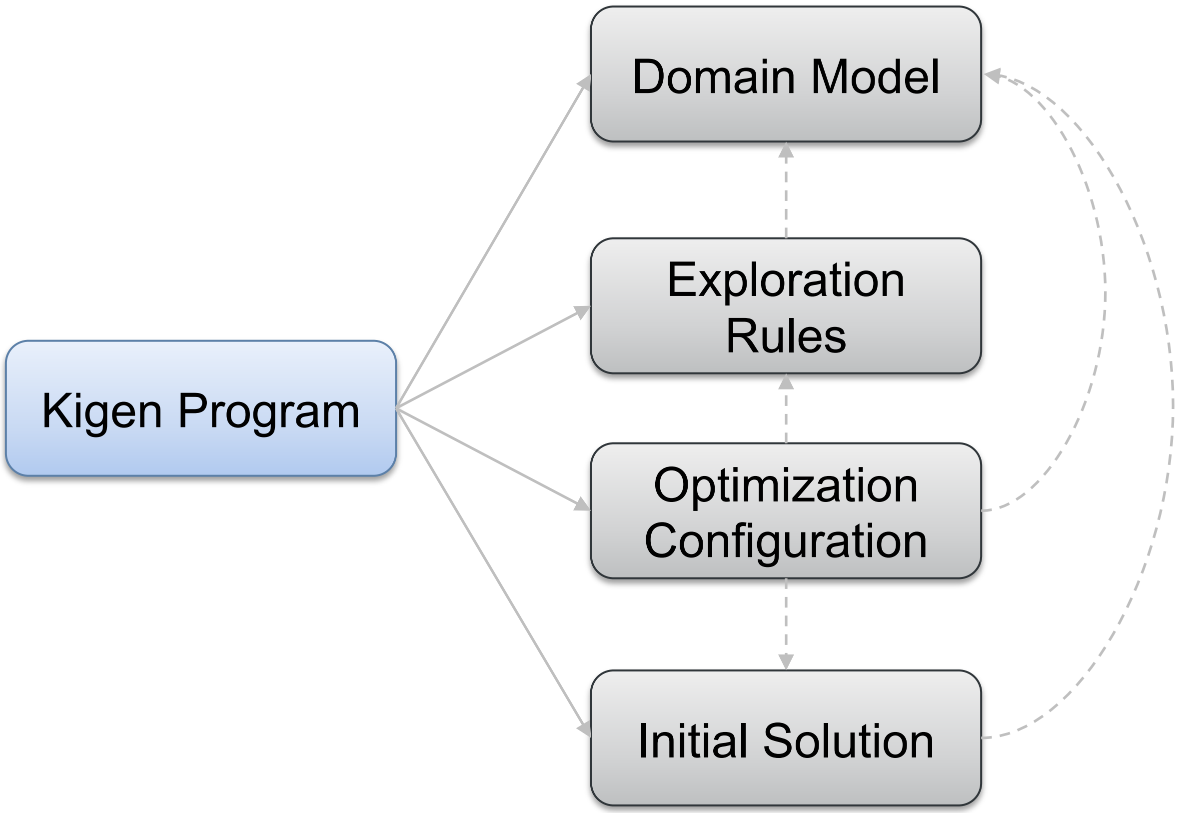
jpl.nasa.gov

Government sponsorship acknowledged. All technical data was obtained from publicly available sources and / or is fictitious.

Jet Propulsion Laboratory
California Institute of Technology

Backup Slides

Derived Artifacts from a Kigen Program

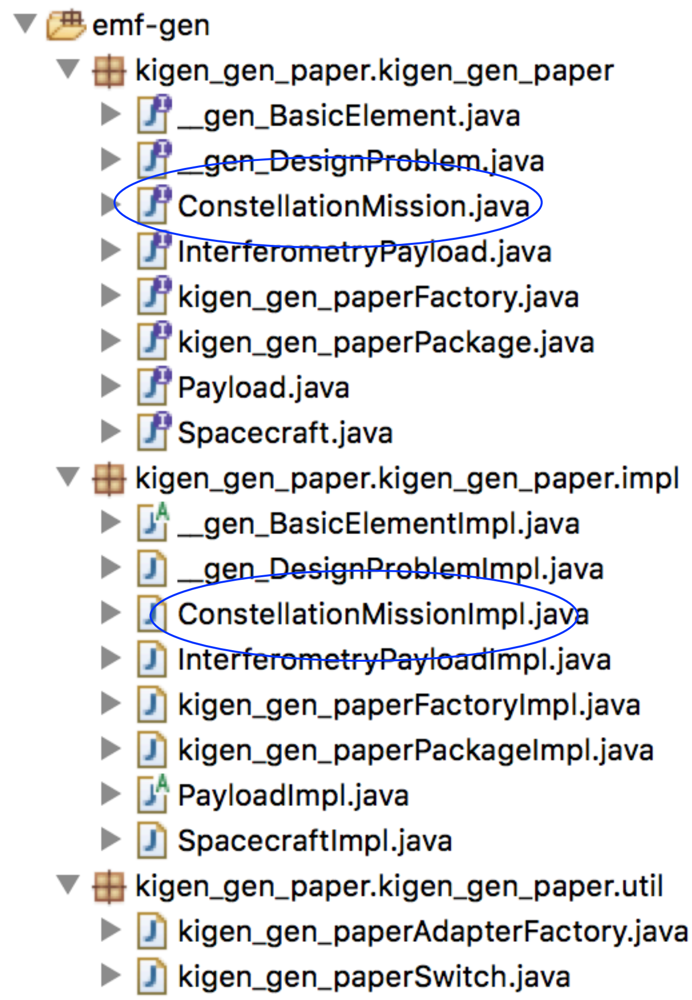


Generated Code

- A Java class is created for each "type" – code generation is based on GenModel mechanism in EMF
- These classes can be directly used in externally defined simulation models

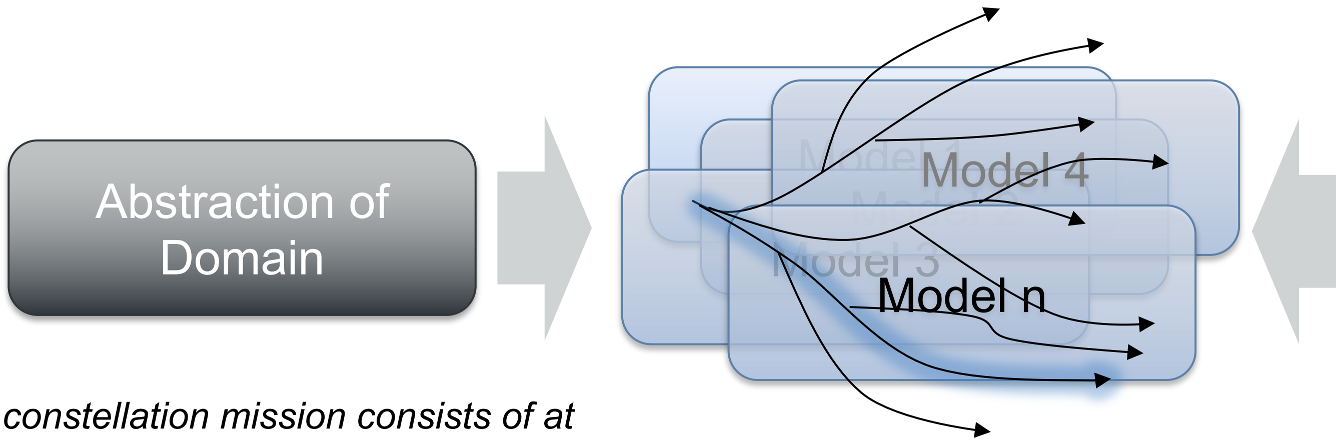
```
// KigenML Program
type ConstellationMission {
    …
    fun someComplexSimulation =
        jvmcall mysim.Sim.simulate(this)
    …
}

// Java Program
public class Sim {
    public static double simulate(
            ConstellationMission theMission) {
        return 0.0;
    }
}
```



Finding Models in Domain

Mechanized Exploration



Which interferometry missions are optimal with respect to cost & scientific benefit?

"A constellation mission consists of at least 2 spacecraft and at most 100"

"A spacecraft can, but does not have to contain the interferometry payload"

"Operation of the interferometry payload operation requires power"

Solution Search
Generation
Best-Performing
Models in domain
Models in domain
E.g.: "Constellation mission A with 3 spacecraft, one of which h

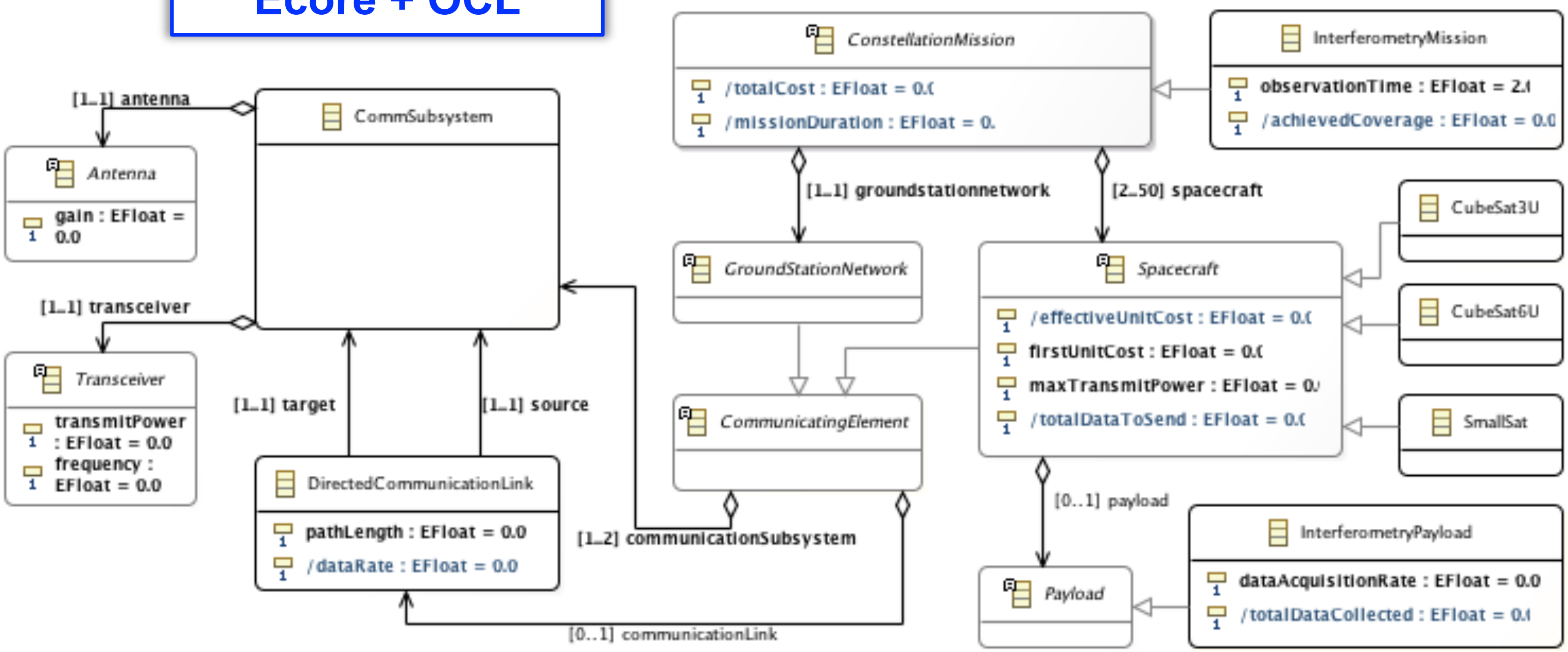
Problem Description
Which models in the domain are we looking for?

In practice, too many possible solutions to generate & compare all
➔ **View as a search problem**

Application to Case Study

Representation of Domain (Excerpt)

Domain model in Ecore + OCL



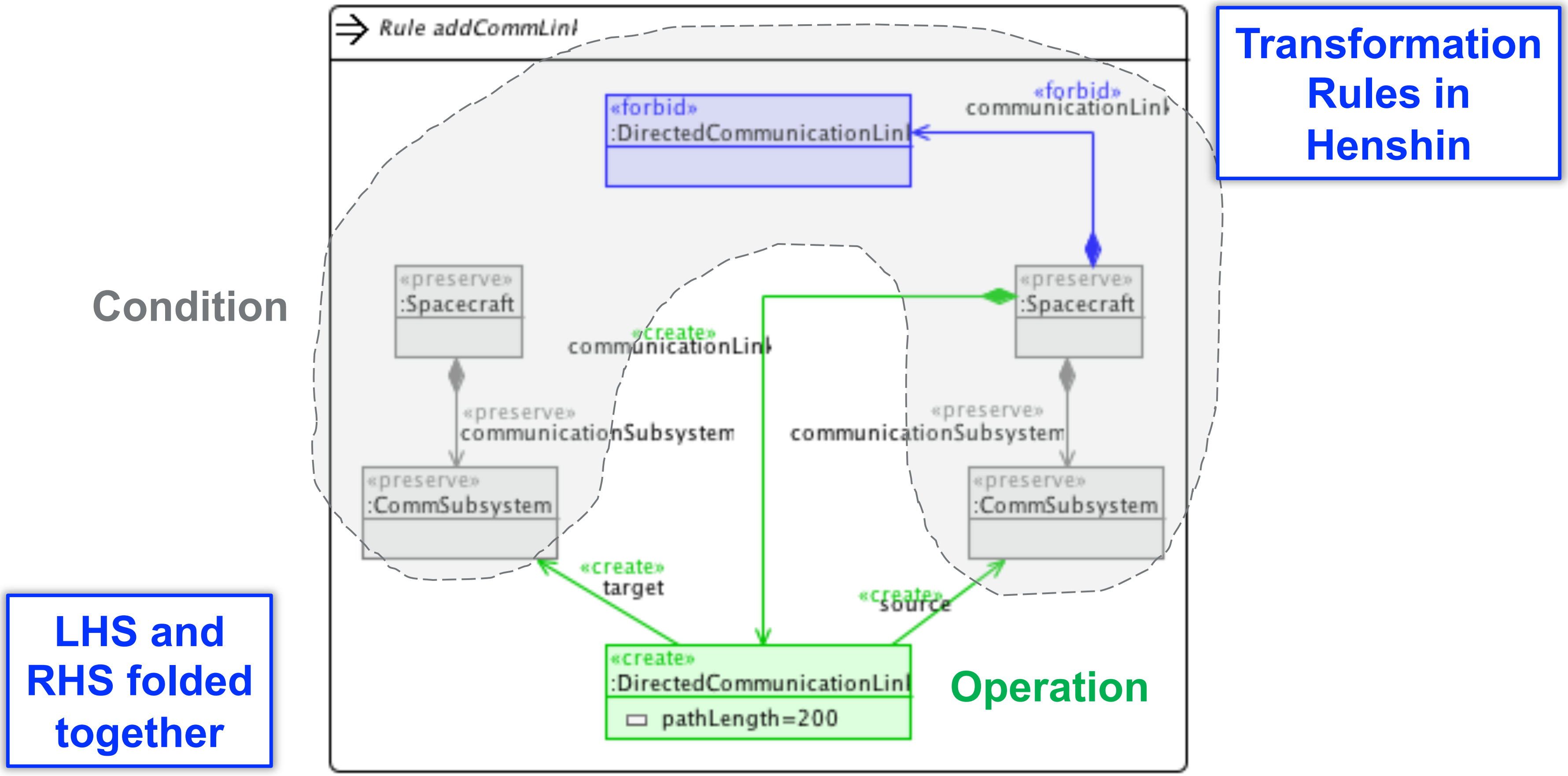
20 concepts, 9 associations, 15 attributes / parameters

> 48^{10} possible models

Too many for exhaustive search

Application to Case Study

Transformation Rule Example (Henshin Syntax): Add Comm. Link

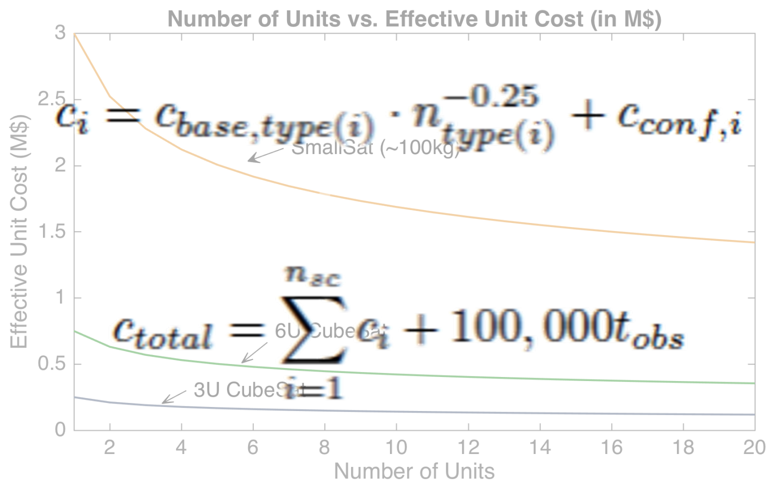


In Prose: *"Find 2 distinct spacecraft instances, and add a communication link between them"*

Application to Case Study

Analysis Model Definitions

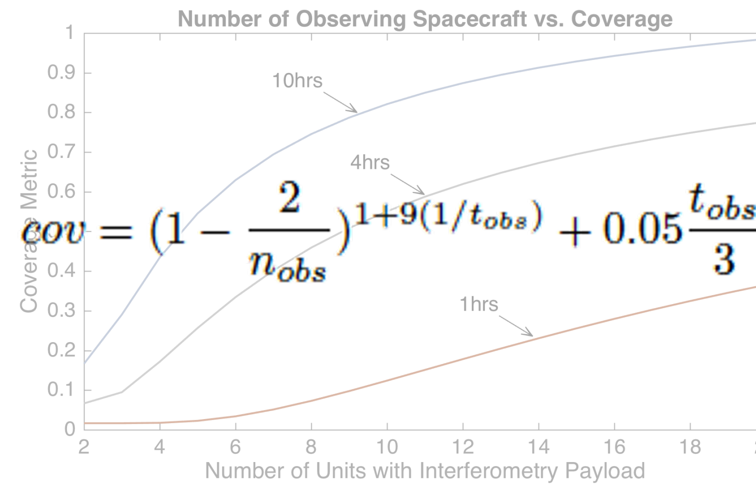
Cost Model



$$c_i = c_{base,type(i)} \cdot n_{type(i)}^{-0.25} + c_{conf,i}$$

$$c_{total} = \sum_{i=1}^{n_{sc}} c_i + 100,000 t_{obs}$$

Science Model



$$cov = (1 - \frac{2}{n_{obs}})^{1+9(1/t_{obs})} + 0.05\frac{t_{obs}}{3}$$

Data Management Model

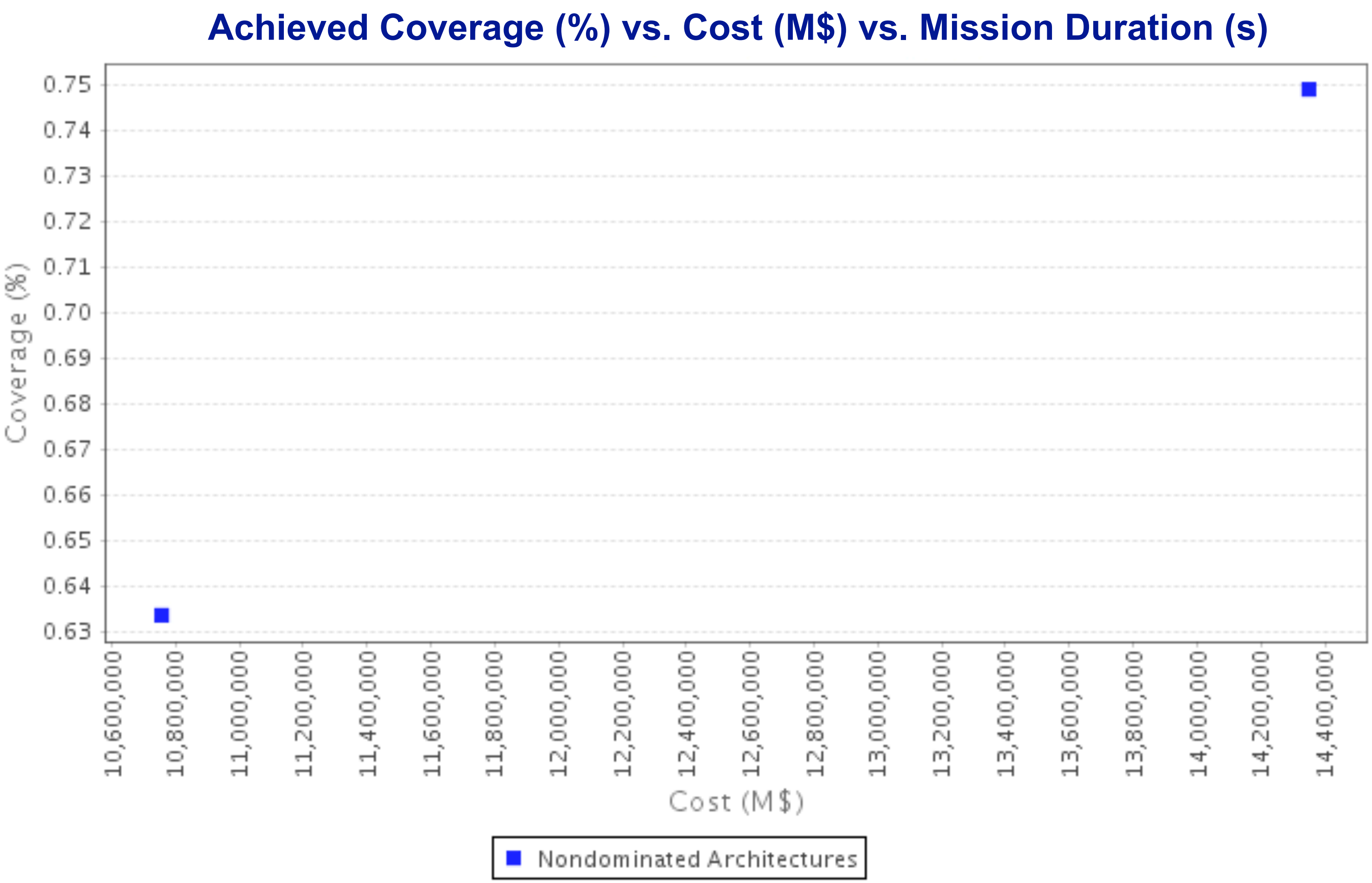
- Data out = own science + data in
- Simplified operations

$$d_{out,i} = d_{science,i} + \sum_{j=1}^{n_{sc}} d_{in,j}$$

Communication / Link Model

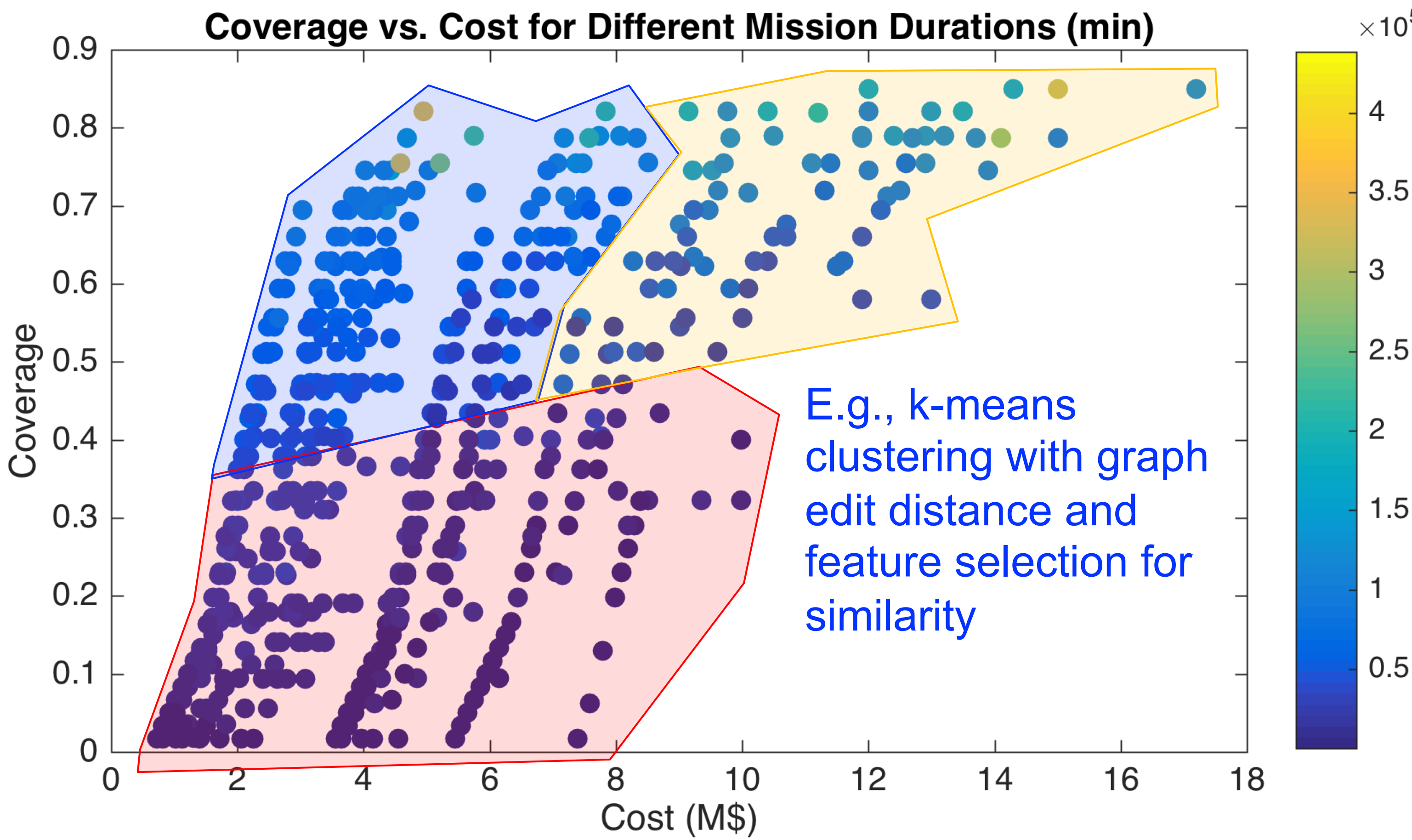
Transmitter Configuration	200 km	385k km
UHF, 3 W, 1 dBi	5 Mbps	-
X-Band, 5 W, 10 dBi	1.6 Mbps	0.7 Mbps
Ka-Band, 15 W, 25 dBi	220 Mbps	80 Mbps

Evolution of Population (Algorithm: NSGA-II)



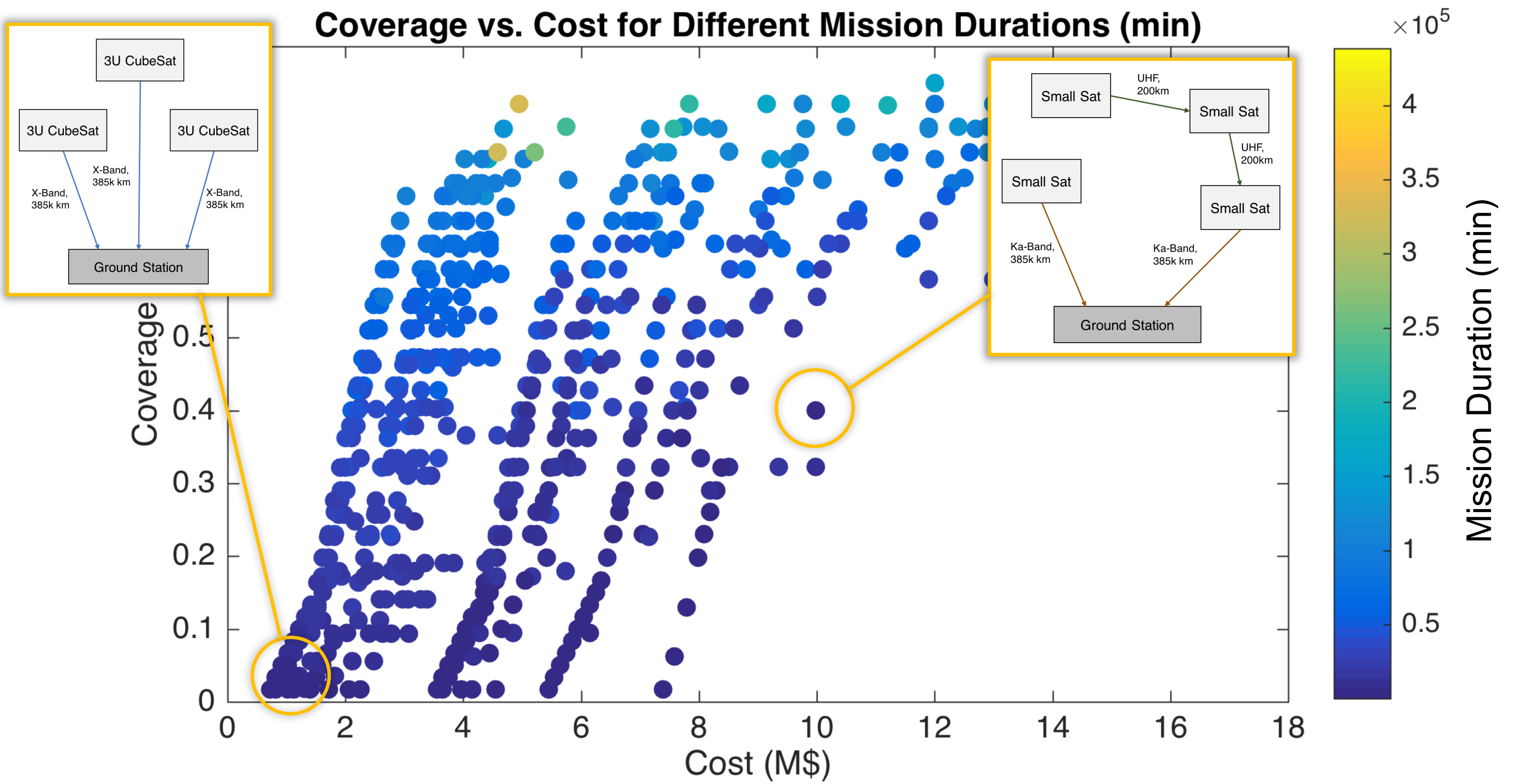
Additional Analysis

Clustering of Solutions



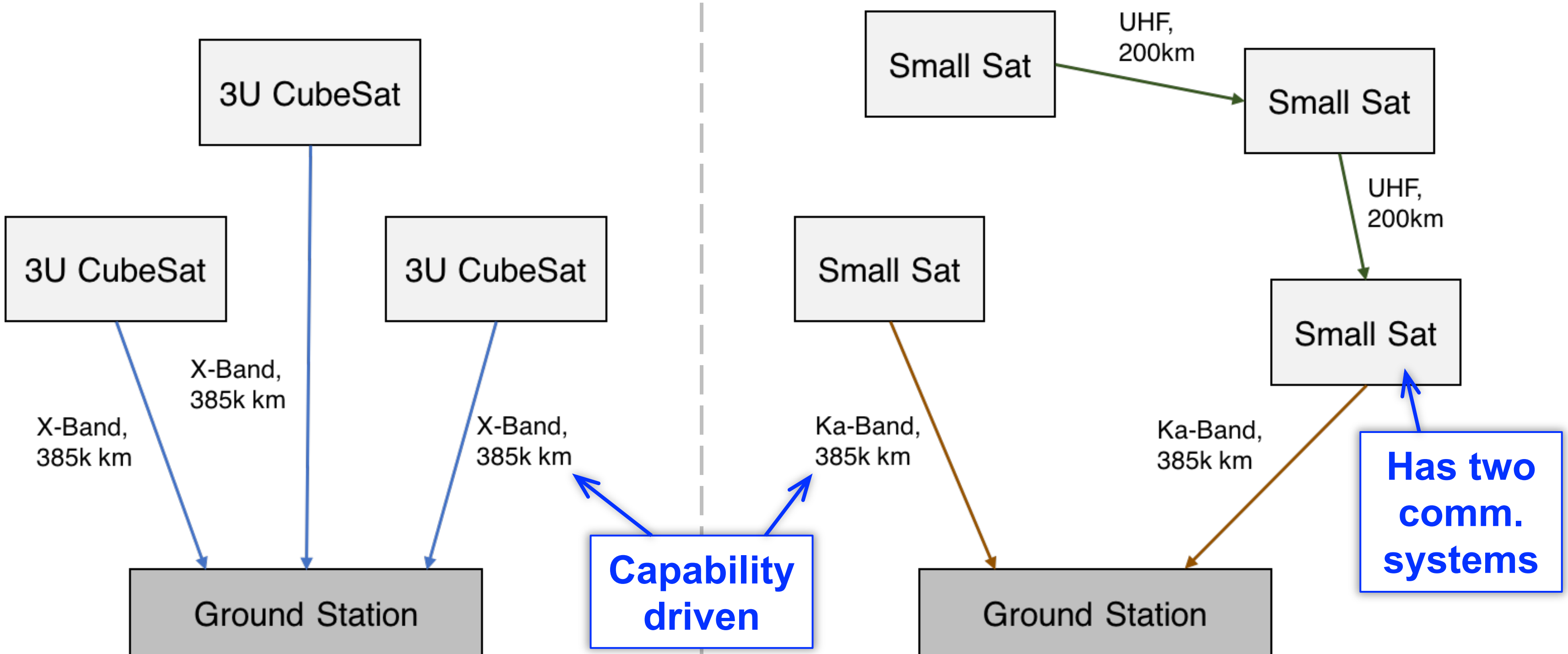
Results from Application to Case Study

Visualization of Trade Space



Results from Application to Case Study

Examples of Pareto-Optimal (Nondominated) Solutions



Candidate Solution #1
$1M, ~0.02 coverage (1h obs.)

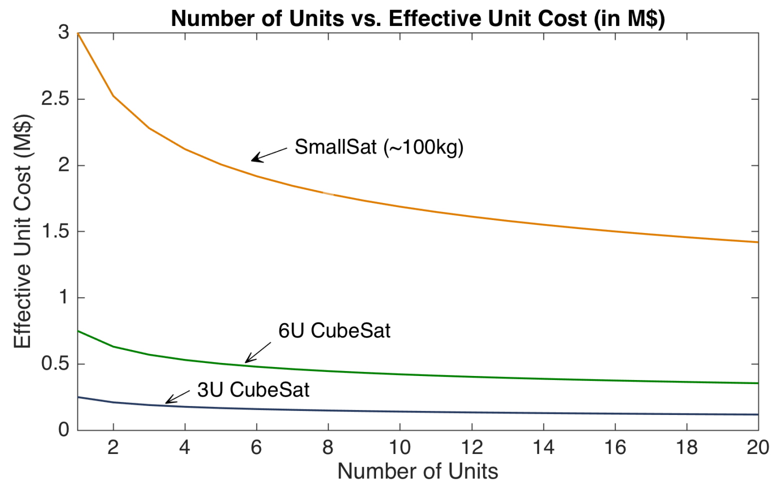
Candidate Solution #2
$10M, ~0.4 coverage (9h obs.)

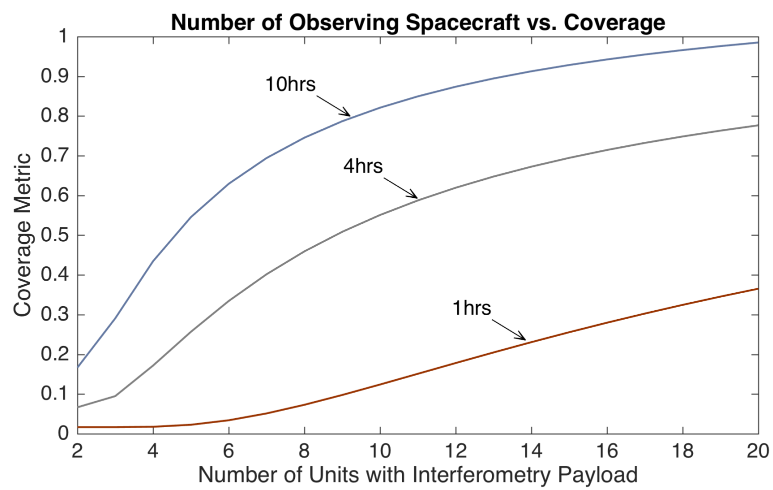
Similar mission duration, but #1 has much longer downlink time

Application to Case Study

- Three objectives:
  - Minimize cost
  - Maximize coverage (measure of scientific benefit)
  - Minimize mission time
- Typical link budget for data rates
- Data collection & transfer model
- Abstracted away orbit design through coverage model
- Experiment setup:
  - 16 transformation rules
  - 180 variables per individual
  - NSGA-II with population size 1000, and 100 generations
  - 30 runs*

* 8 core Intel i7 @ 2.7Ghz, 16GB DDR3 RAM

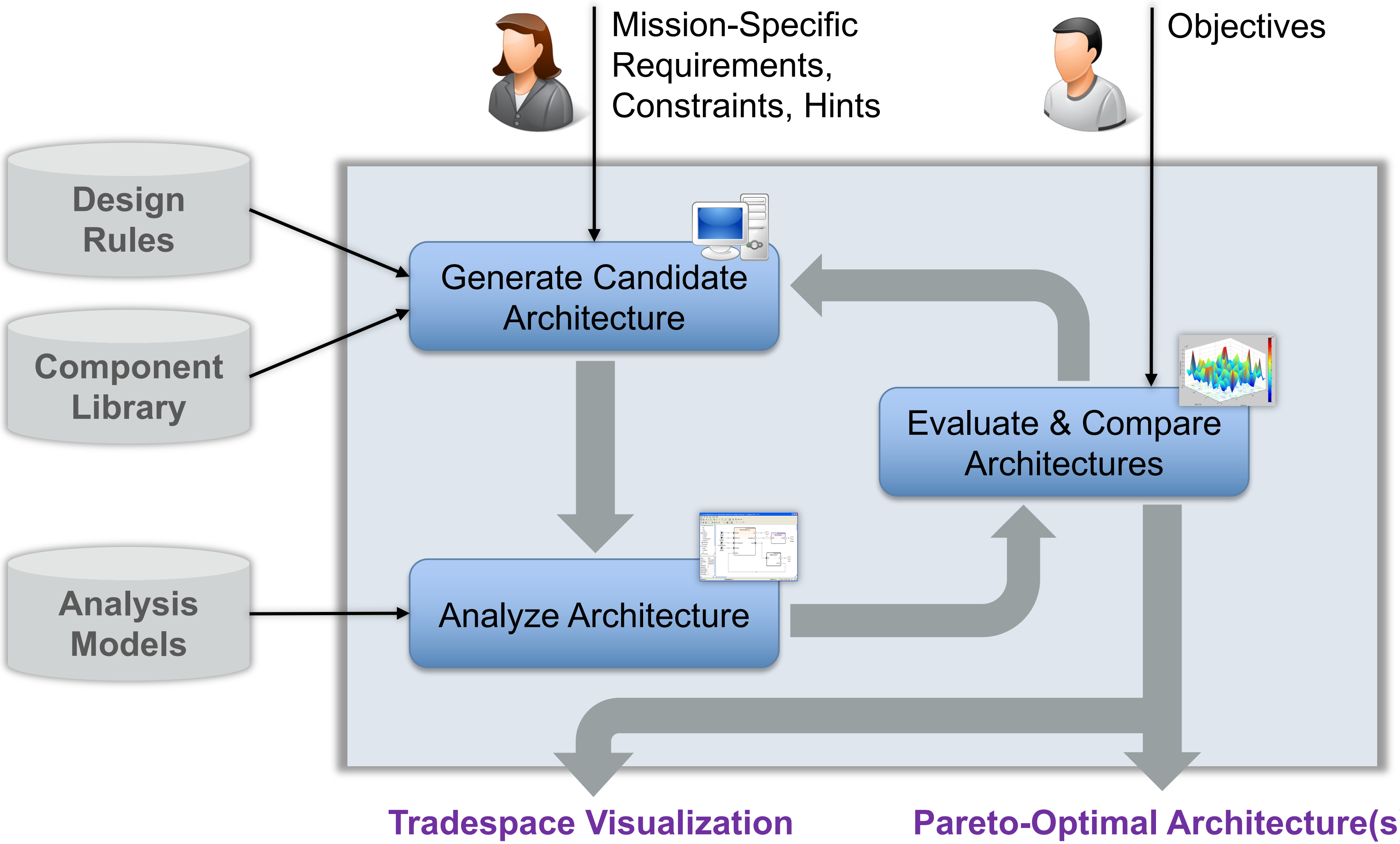




<u>Fictitious</u> cost model (top) and coverage model (bottom)

Framework

CDS for Mission Architecture Design



Application to Case Study

Link Calculations

- Derived from standard link budget, assuming above average noise due to expected interference from Moon

Table 1. Computed communication rates. 385k km case assumes 72 dBi receive antenna gain for X-band, and 85 dBi for Ka-band (similar to DSN).

Transmitter Configuration	200 km	385k km
UHF, 3 W, 1 dBi	5 Mbps	-
X-Band, 5 W, 10 dBi	1.6 Mbps	0.7 Mbps
Ka-Band, 15 W, 25 dBi	220 Mbps	80 Mbps

Application to Case Study

Cost Calculations

- Cost per spacecraft calculation incorporates a learning curve
- Assuming $ 100,000 per hour of observation to estimate observation and data processing cost

$$c_i = c_{base,type(i)} \cdot n_{type(i)}^{-0.25} + c_{conf,i} \tag{5}$$

$$c_{total} = \sum_{i=1}^{n_{sc}} c_i + 100,000 t_{obs} \tag{6}$$

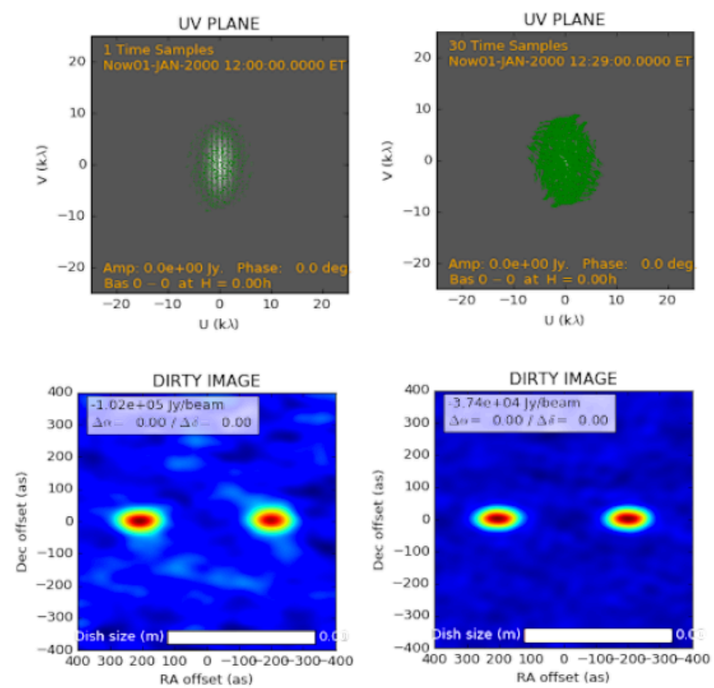
Application to Case Study

Coverage

- Simple coverage calculation

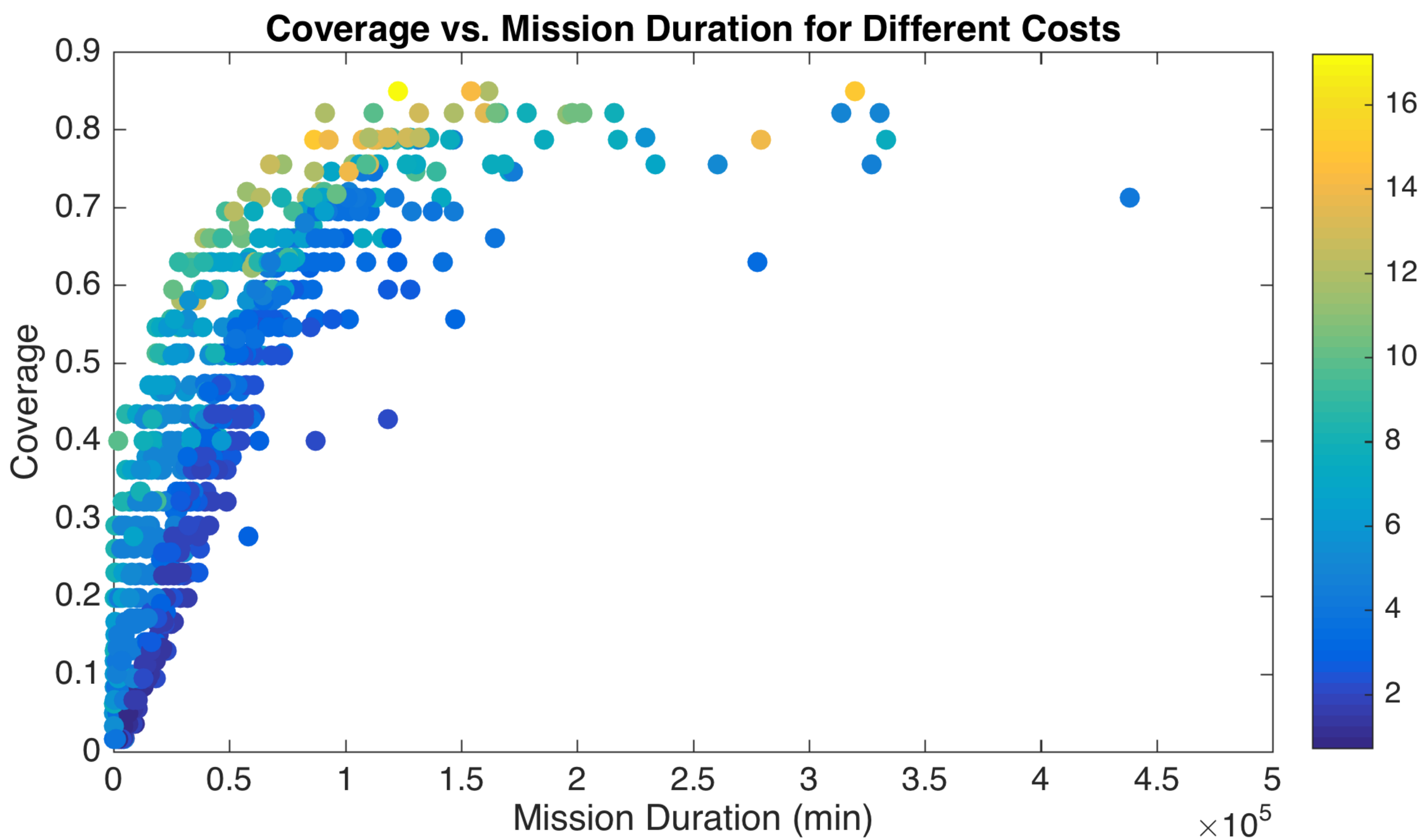
$$cov = (1 - \frac{2}{n_{obs}})^{1+9(1/t_{obs})} + 0.05\frac{t_{obs}}{3} \qquad (1)$$

- Surrogate model that reflects trends observed from more sophisticated telescope array simulation performed by Alexander Hegedus (https://github.com/alexhege/Orbital-APSYNSIM/)



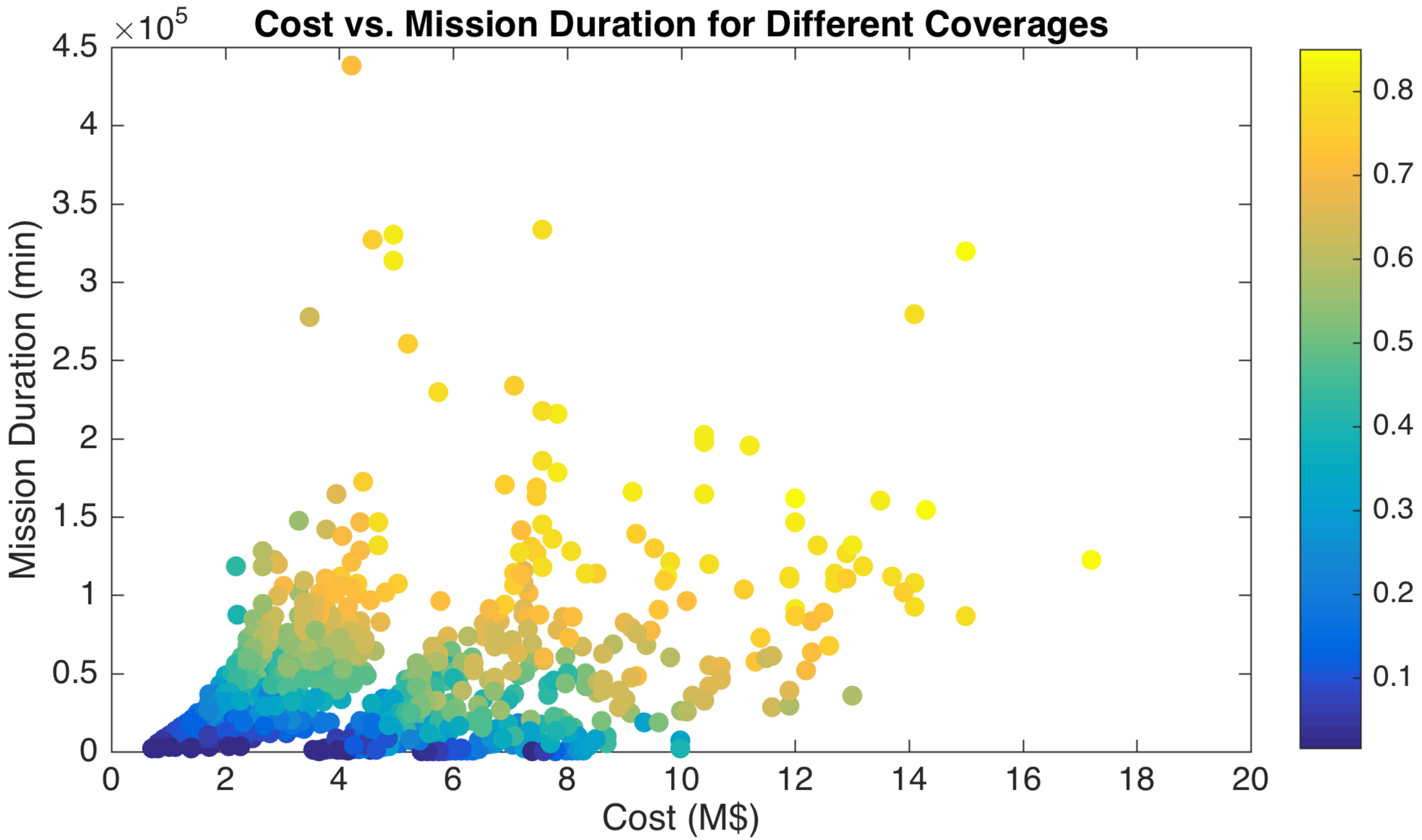
Results from Application to Case Study

Coverage vs. Mission Duration



Results from Application to Case Study

Cost vs. Mission Duration



Motivating Case: Mars Cave Exploration

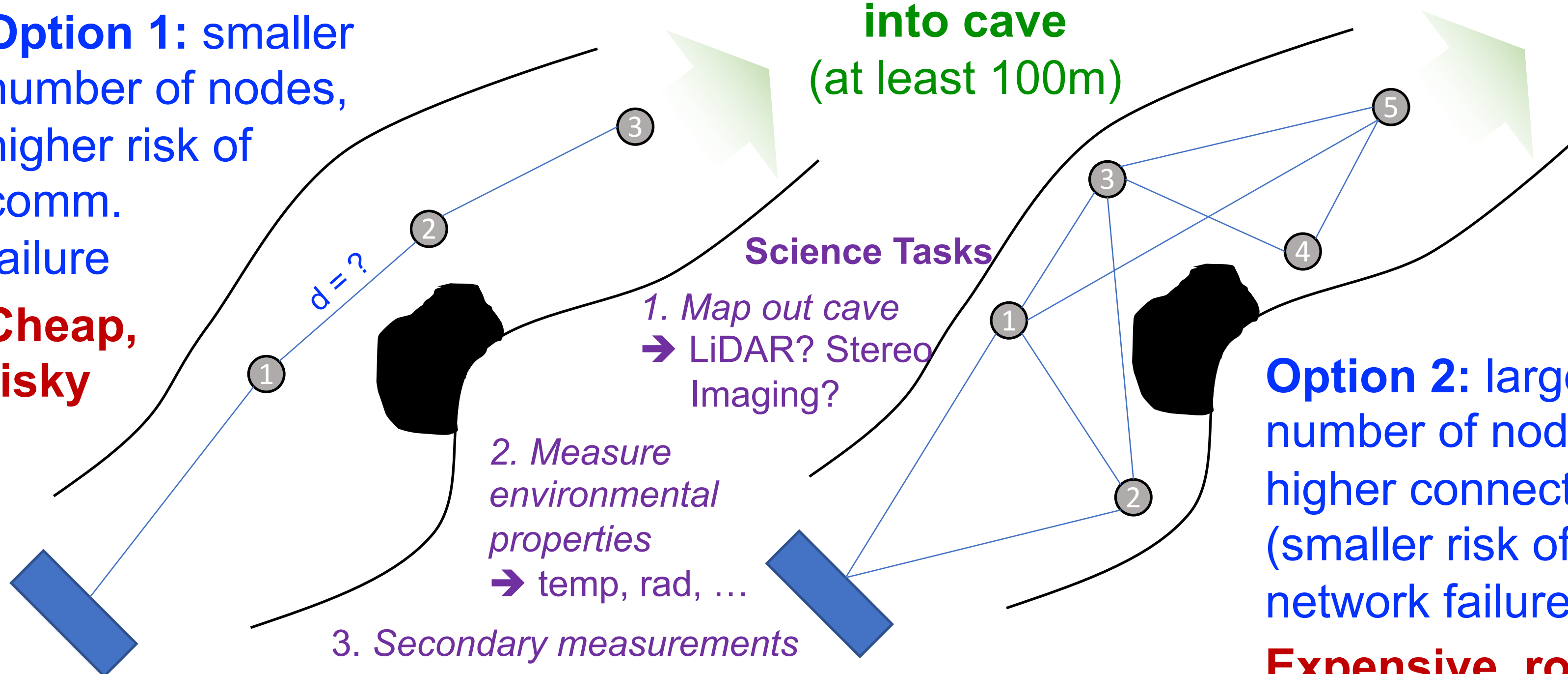
Select Network-, Vehicle- and Operations-Level Trades

Minimize cost, risk

Maximize distance into cave
(at least 100m)

Option 1: smaller number of nodes, higher risk of comm. failure

Cheap, risky



Science Tasks

1. *Map out cave*
➔ LiDAR? Stereo Imaging?

2. *Measure environmental properties*
➔ temp, rad, …

3. *Secondary measurements*

Maximize science return

Option 2: larger number of nodes, higher connectivity (smaller risk of network failure)

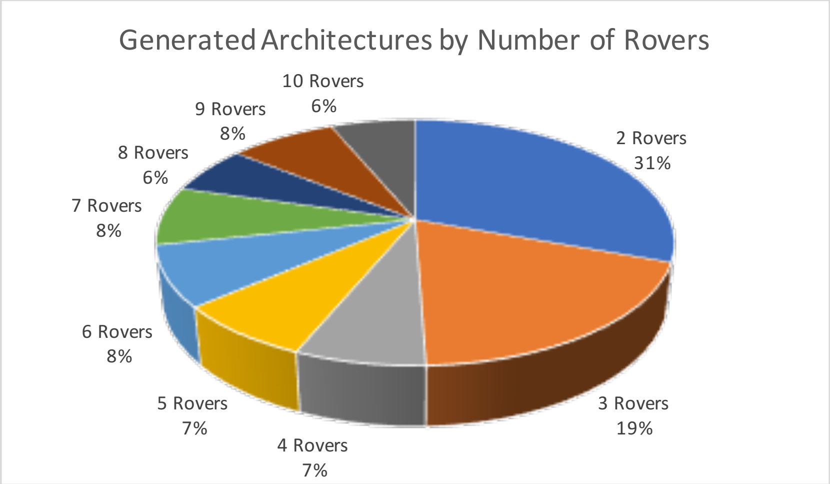
Expensive, robust

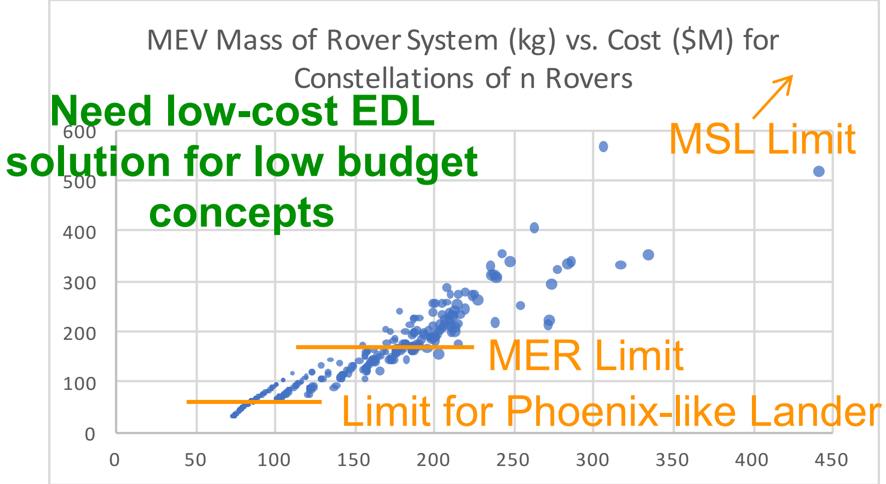
Small spacecraft ~ limited capabilities
➔ **How many rovers? Which assets are equipped with which instruments?**

Obstacle
Spacecraft Vehicle
Base (e.g., Axel rover)
Possible comm. path

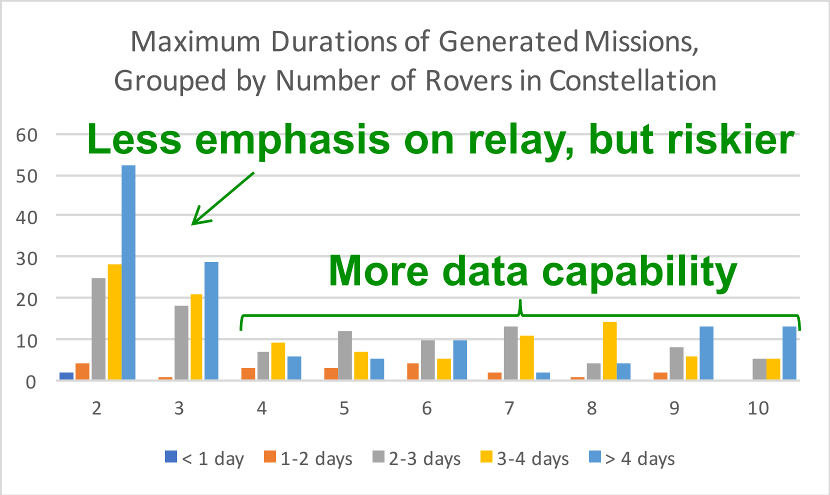
Analysis & Visualization of Optimizer Results

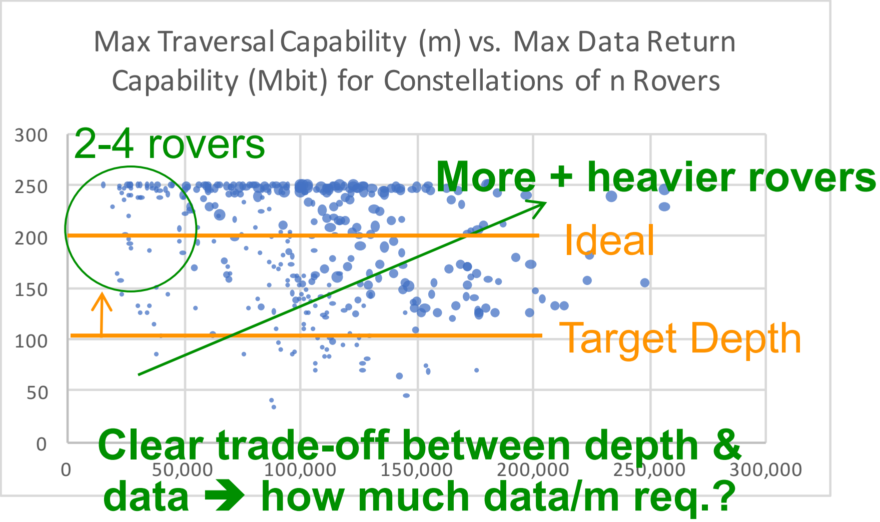
Analysis of Feasible Solutions (l) & Visualization of Trade Space (r)



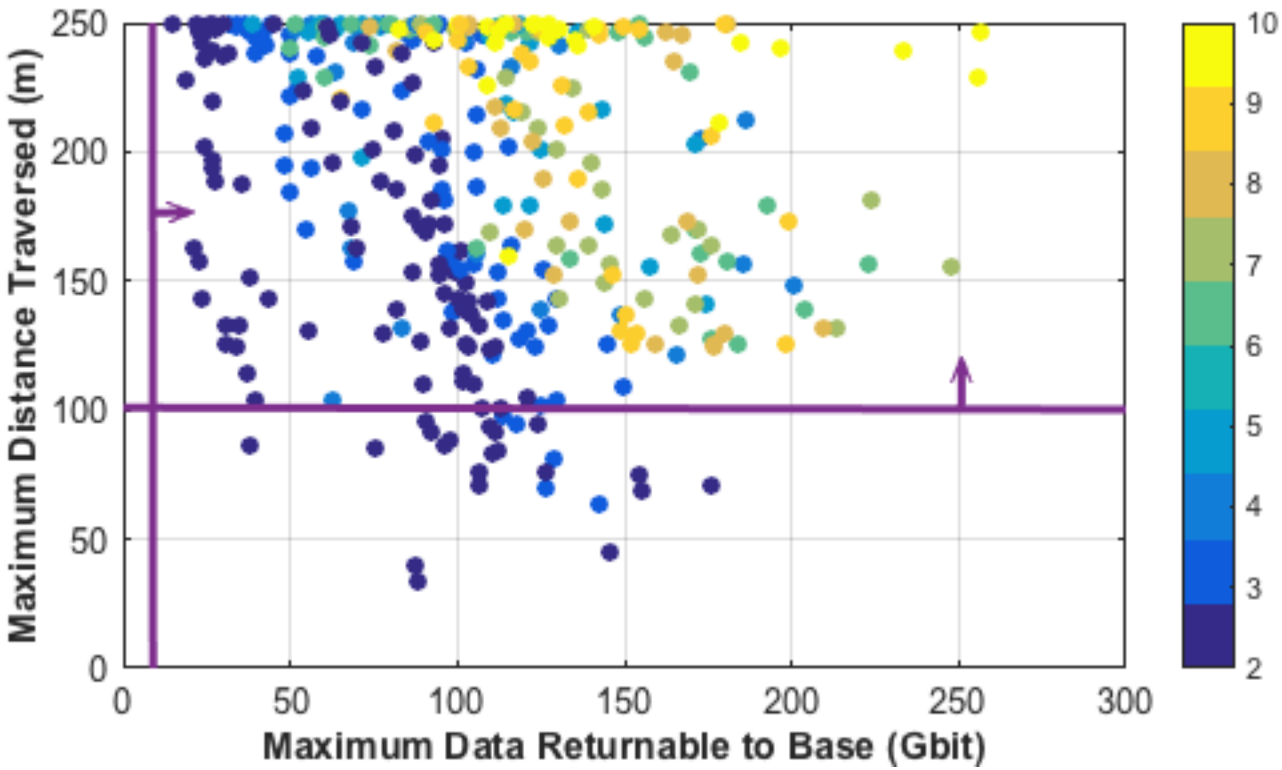


➔ Integrated vehicle, operations & network design optimization



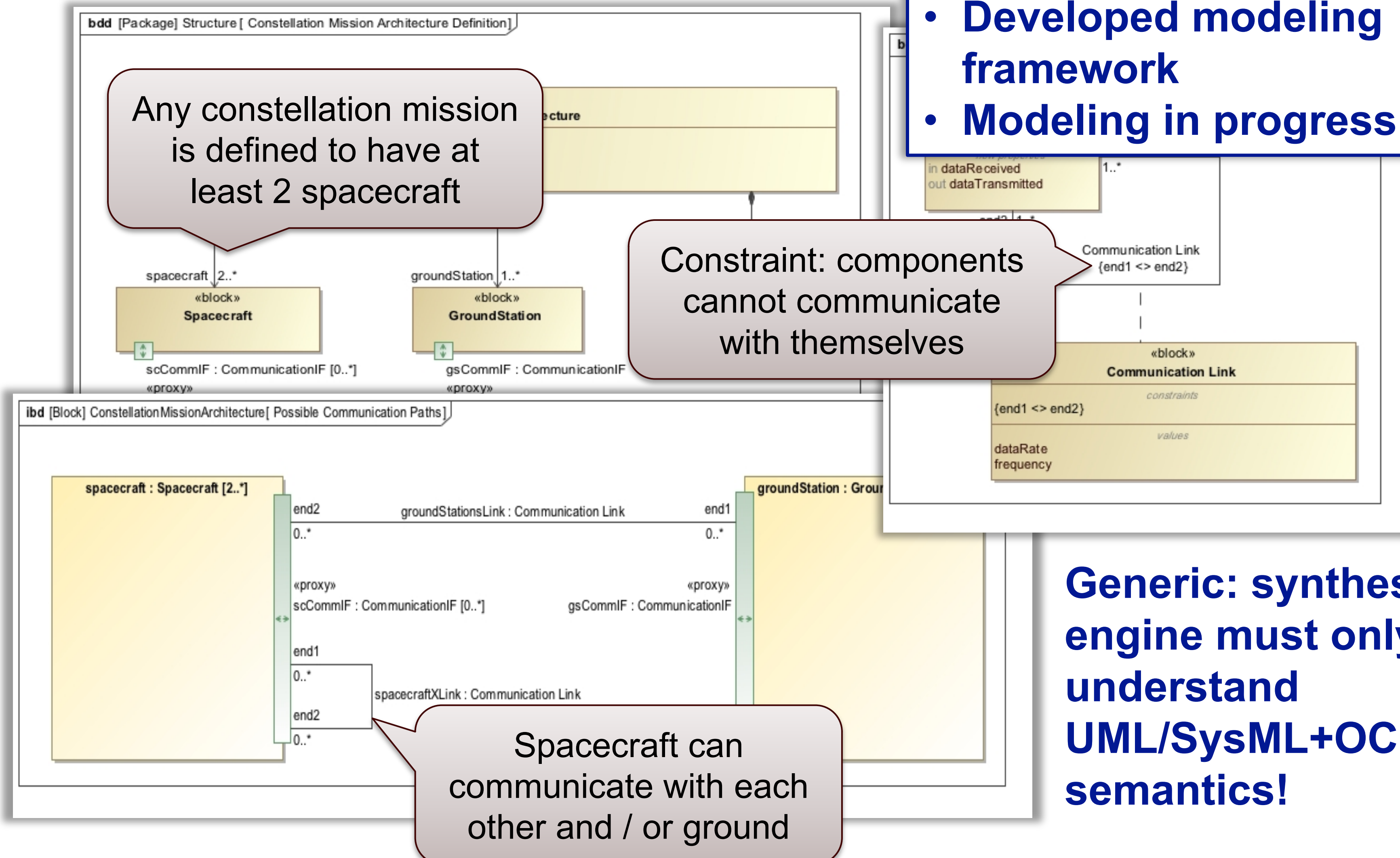


Results from Mars Cave Exploration Mission Case Study



Capturing Design Options & Rules Formally

Optimized Integrated Network Constellation Design

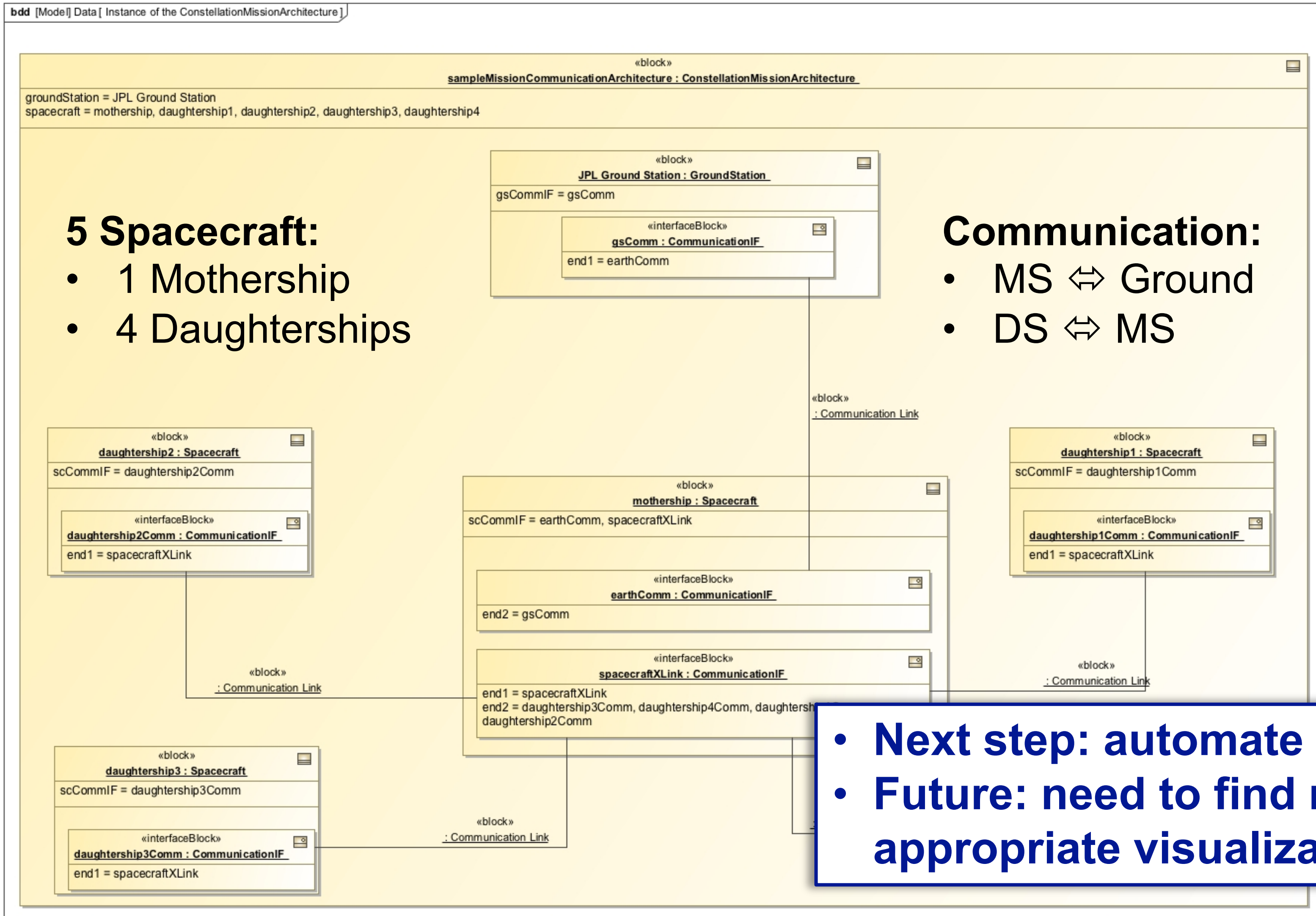


- **Developed modeling framework**
- **Modeling in progress**

Generic: synthesis engine must only understand UML/SysML+OCL semantics!

Instance Creation: Example Partial Instance

Optimized Integrated Network Constellation Design



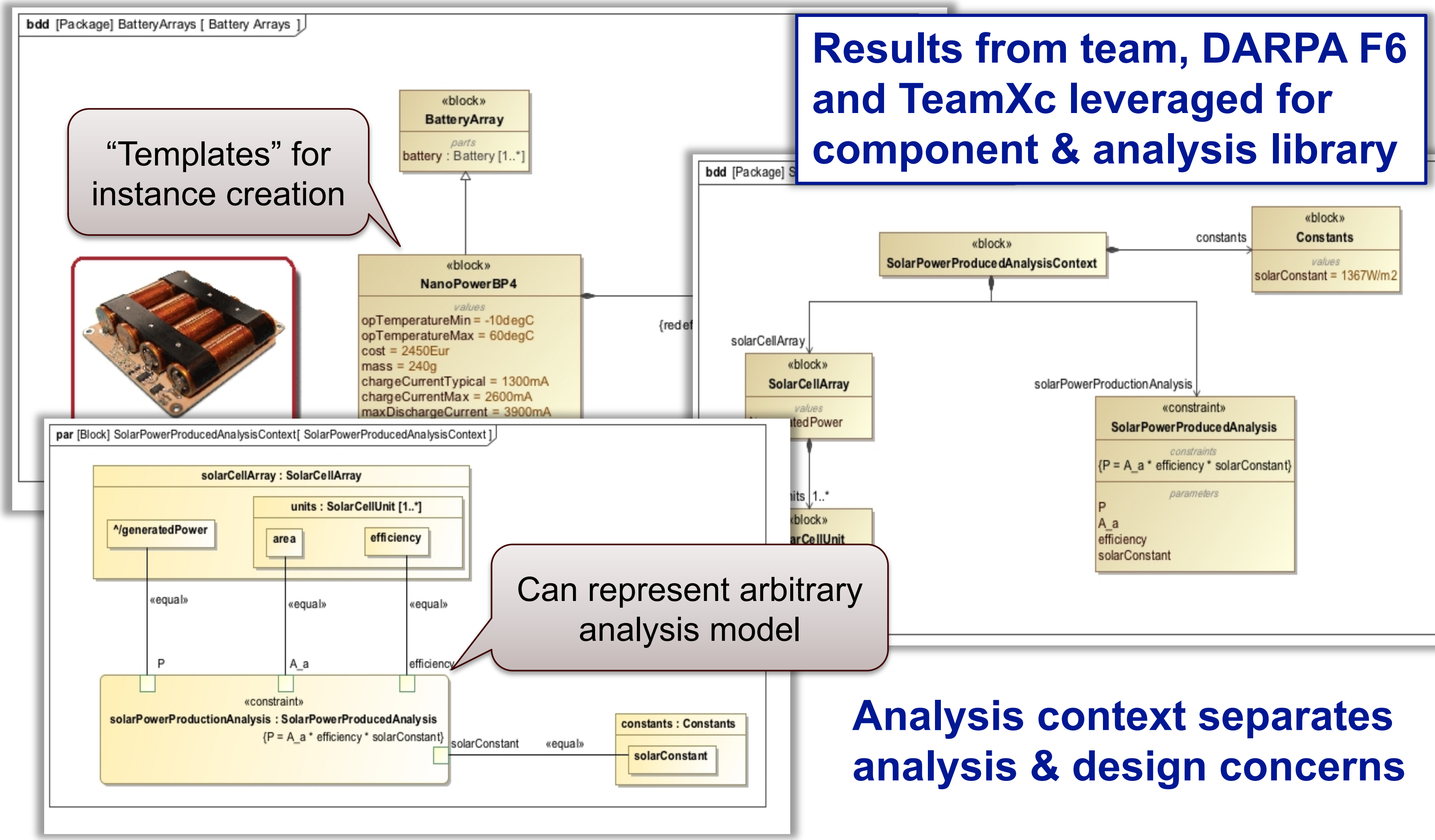
5 Spacecraft:
- 1 Mothership
- 4 Daughterships

Communication:
- MS ⇔ Ground
- DS ⇔ MS

- **Next step: automate**
- **Future: need to find more appropriate visualization**

Component & Analysis Library

Optimized Integrated Network Constellation Design



“Templates” for instance creation

Results from team, DARPA F6 and TeamXc leveraged for component & analysis library

Can represent arbitrary analysis model

Analysis context separates analysis & design concerns

Content / Outline

• Context / problem statement	2min	[p]
• Challenge	1min	[r]
• Approach		
– Rule-based exploration, meta-model + rules, chain of rules	2min	[a]
– KigenML to specify, include objectives	1min	[n]
– Optimization mechanism(s) (+ mention post-processing)	1.5min	[a]
• Results		
– Interferometer	1min	[r]
– [Cave]	1min	[r]
– (Just mention clustering)		
– (What was surprising?)		
• Conclusions	1min	[n]